UNIVERSITÀ DI PISA

Facoltà di Scienze Matematiche, Fisiche e Naturali

Dipartimento di Fisica

Tesi di Laurea Magistrale in Fisica

A ground state search algorithm for strongly correlated 1D quantum systems

Candidato:
Alessandro David

Relatore:
Prof. Vittorio Giovannetti

Correlatore:
Dr. Davide Rossini

ANNO ACCADEMICO 2014/2015

To my wife Noemi
whom I love and cherish

Acknowledgements

I am deeply grateful to my supervisors and I wish to thank them for the opportunity they gave me to develop their idea. I have appreciated all their advices and support and I truly enjoyed to work with them on this project.

I also want to express my sincere gratitude to my wife for all the help and strength that she gave me to finish this thesis and that she always gives me to live everyday with happiness. Thank you Noemi, I love you so much.

An important thought goes to all the people that helped me become who I am today. The list is too long and surely I will forget someone, but I want to remember all the friends that I met in Trento: my great flatmates, Roberto and Riccardo, and all the physicists of my year. I also want to remember my dear friends from high school: Anna, Fabio, Alain and Ross, and my sister Giulia. Last but not least, I give a huge hug to the girls that helped me and Noemi survive in our new adventure in Pisa, some time ago already: thank you Serena, Fabiana and Martina. What can I say, you are all wonderful people.

Finally, I would like to thank my parents that sustained me and backed me up for every activity I wanted to do and, most importantly, they patiently listened to every explanation of Physics I wanted to share with them, no matter how puzzling. That was surely a sign of great love for me.

Contents

List of Abbreviations

AdS/CFT : Anti de Sitter / Conformal Field Theory.

AGSP : Approximate Ground State Projection.

AKLT : Affleck, Kennedy, Lieb, Tasaki.

CK : Chebyshev Kicker.

DMRG : Density Matrix Renormalization Group.

FCS : Finitely Correlated States.

HK : Hamiltonian Kicker.

ITEK : Imaginary Time Evolution Kicker.

LHVM : Local Hidden Variables Model.

LHS : Left Hand Side.

LOCC : Local Operations and Classical Communication.

MPO : Matrix Product Operators.

MPS : Matrix Product States.

NRG : Numerical Renormalization Group.

OBC : Open Boundary Conditions.

PBC : Periodic Boundary Conditions.

RHS : Right Hand Side.

SVD : Singular Value Decomposition.

Introduction

One of the reasons physics is a very hard subject is that with many real-life problems we need to describe an enormous amount of parameters. Sometimes we are overwhelmed and defeated and the problem is not solvable, some others we are lucky and an underlying symmetry or a mathematical theorem help us win the battle. For example, the purpose of thermodynamics and statistical mechanics is to find those laws and relations that allow us to reduce and control this mind blowing quantity of data with much fewer values. The situation gets more complicated passing from classical to quantum systems: the number of parameters (that describe the state) no longer scale linearly with the number of particles, but exponetially. This is a direct consequence of the tensor product structure of composite quantum systems. Usually, we try to simulate these systems numerically, but, increasing the size, very soon we hit a wall and we have to stop for lack of resources. Under these circumstances there seems to be no hope to be able to reliably investigate quantum many-body systems with a computer. Feynman was aware of this issue and he theorized that, to simulate efficiently a quantum system, it is better to use another quantum system [1]. Nevertheless, we will not talk about quantum simulations (an important topic of quantum information by itself); in fact, this work is based on a recent discovery that drastically simplifies the structure of ground states in one dimension, so that they are still treatable with a classical computer.

We focus on systems whose constituents are finite dimensional (spins) and reside on the sites of a lattice. The hamiltonians that govern these systems contain local and interaction terms, such as nearest-neighbor interactions, second-nearest-neighbor interactions and so on. They can be seen as effective hamiltonians that abstractly describe the relevant physics of a real system. Anyway, recent advances in the field of quantum optics allow the creation of quite arbitrary and tunable hamiltonians of this kind, introducing ultracold and dilute atom gases in optical lattices [2]. This is one of the reasons why theoretical interest in the field has increased in the past years.

From a historical point of view, early attempts to solve analytically these

kinds of problems can be traced back to the introduction of the Heisenberg model and its solution [3, 4], to the quantum version of the Ising model[1] and to the Bethe ansatz [6]. The latter is one of the very few tools that, in one dimension, allows to find some answers. Perturbation theory fails here because these systems present very strong interactions. The first good numerical method was the Numerical Renormalization Group (NRG), proposed by Wilson in the 1970s, that successfully solved the Kondo problem [7]. Unfortunately, this approach was working only for small quantum impurities in a classical lattice and not for full quantum spins.

Addressing some of the weaknesses of NRG, White introduced the Density Matrix Renormalization Group (DMRG) in 1992 and it still is the best instrument to search ground states of strongly correlated systems that we have[2]. DMRG came in two flavours, one for infinite systems, where the size of the system grows continuously adding new sites and minimizing the energy every step, the other for finite systems, where the system is swept back and forth minimizing the energy at every site of the lattice. Very soon, for both of these versions, it was recognised that the states produced in the process were, actually, Matrix Product States (MPSs) ([9] for the infinite version and [10] for the finite version). MPSs are a particularly attractive form to write a many-body quantum state as the product of many matrices, but for some time their connection with DMRG was disregarded. The reasons why this connection reemerged are very important.

A first reason is linked with the fact that the success of DMRG is not universal: its best performance is only in one dimension. For some time it was unclear why, but the explanation came from quantum information and entanglement theory in the early 2000s (see e.g. [11] or [12]). The so-called "area laws", that we will discuss in details, offer an elegant theoretical justification and a remarkable simplification for DMRG in one dimension. They state that ground states of local, gapped hamiltonians possess relatively small entanglement and we will see how this means relatively few parameters to process. The importance of MPSs enters here because they are objects with a built-in area law and thus they constitute the optimal class of states where to look for the ground state.

A second reason is that, in the same years, Cirac, Verstraete, Vidal *et al.* began to reformulate DMRG in MPS language and they found many new algorithms, hard to notice in a pure DMRG approach. Among them we cite real-time evolution [13] and finite-temperature algorithms [14]. Today, the

[1] For a complete review, see [5].

[2] Another possible method is the quantum Monte Carlo, but it suffers a sign problem for fermionic particles [8], which the DMRG does not experience.

theory of MPSs in one dimension is well developed and an excellent review is [15], we will use it extensively.

The original contribution, in our work, is the development and implementation of a new algorithm for ground state search that, like the procedures cited above, exploits the MPS language as an economical way to express very large vectors. The idea, inspired by the Lanczos algorithm for eigenvalues [16], is to start from a random vector and apply to it an operator K, called the *kicker*. Now we possess two vectors that span a two-dimensional subspace; we minimize the energy of the reduced hamiltonian for that subspace and we generate the vector that points in that direction. This strategy always finds lower or equal energy values, because the initial vector is not excluded from the possible choices. We apply the same process to this new vector and we repeat so until the energy converges.

A lot of work has been dedicated to the research of the optimal kicker K: we compare and list advantages and disadvantages of at least three different classes of kickers such as the hamiltonian kicker, the imaginary time evolution kickers and the Chebyshev kickers. The latter comes from an idea expressed in [17]; the first two, instead, are quite ordinary. No optimal solution was found, though, because while the hamiltonian always results in a good kicker, the performance of the other two types could be better, but it is very problem-dependent. We tested our kickers on the Ising model and we compared them with DMRG.

Since we do not minimize the energy site by site as in DMRG, but we change completely the state vector at every step, we hope to find novel behaviours that will improve the performance of ground state search. Indeed, we will see that this method has a very fast initial energy descent that, especially with very large systems, is faster than DMRG, but it also has a very long asymptotic convergence at the end. Finally, we propose to create an hybrid algorithm, where we begin with the new algorithm, initially faster, and at some point we switch back to DMRG, that is optimal at the end.

A major effort in this work has been the writing of the program that implements the algorithm suggested here. This program includes a small library of functions to work with an arbitrary number of MPSs (and their generalization to operators, MPOs). This effort repaid itself, since the program is now a very versatile tool with which many simulations are possible. The chosen programming language is Fortran 90/95 for the performance with linear algebra routines and it has been compiled with `ifort`, the Intel Fortran compiler freely available for students, and its Math Kernel Library that provided the optimized LAPACK[3] routines. `SlocCOUNT`, a famous program

[3] `http://www.netlib.org/lapack/`

to count the physical lines of code[4], computes more than 4900 lines of code and this is why it was not possible to add the code in appendix, but it is freely available under GNU/GPL licence for anyone who wants it[5].

To summarize, in this thesis, we illustrate our work and findings through the steps described below.

- In chapter 1, we run through some basic notions of quantum information, such as reduced density matrices, Schmidt decomposition and a hint to the theory of entanglement measures; we need all of them to grasp the meaning of the "area laws" expressed at the end of the chapter.

- In chapter 2, we introduce the theory of Matrix Product States as expressed in [15], starting from their definition and some examples. Then we illustrate the gauge freedom in the choice of the matrices that compose an MPS, this is of great help to simplify and implement manipulations of MPSs. We go on to show how MPSs possess a built-in area law if we fix the size of the matrices and how every vector can be expressed in MPS form if we allow the matrices to have arbitrary dimensions. There is a useful graphical notation that represents MPSs visually and that clarifies operations that involve a copious number of indices. At this point we explain all the basic operations with MPSs: overlaps, expectation values, transfer matrices and sums; with the sums we note that the size of the matrices could increase, so we need a method to bring back the dimensions to their original values. Therefore, the two possible kinds of compression follow: compression with singular value decomposition and variational compression. At the end of the chapter, we present the generalization of MPSs to operators, called Matrix Product Operators (MPOs). We explain how to operate with them as they are very important and they come with their own transfer matrices and expectation values. Finally, we show how to convert a typical many-body hamiltonian in MPO form.

- In chapter 3, we first describe the mechanism of the well established algorithm for ground state search: the Density Matrix Renormalization Group, with which we will have to compare our new strategy. We will introduce the DMRG in the language of MPSs and MPOs as included

[4] i.e. pure lines of codes with no comments or empty lines

[5] Email address: `ale.dvdx@gmail.com`.

in [15]. Aferwards, we present the original contributions of this work; those are: the new scheme for ground state search, the analysis of its computational cost, the implementations of the various studied kickers. Among them we find the hamiltonian kicker, that directly uses the hamiltonian as kicker. Then comes the imaginary time evolution kicker that is generated from a truncated Taylor expansion of the exponential of the hamiltonian and last, but not least, there is the kicker based on Chebyshev polynomials and the concept of Approximate Ground State Projection.

- In chapter 4, we talk about the results of the tests that we carried out with the kicker strategy: ground states at various field strengths for the Ising model to evaluate the correctness of the implementation, analysis of the performances of the kickers, comparison with DMRG and finally the performance of the hybrid algorithm (kicker+DMRG).

All the results are commented and reviewed in a final and conclusive chapter, in which we also present our ideas and suggestions for further developments of this new algorithm. There is also a technical appendix A, where we remind the most useful properties of the Singular Value Decomposition (SVD), a factorization of linear algebra that is used many times in MPS theory.

Chapter 1

Quantum information tools and area laws

Quantum many-body systems pose an interesting challenge in modern condensed matter physics because they are notoriously hard to solve. Analytical solutions have been found only for very simple models and often the only way through is numerical. Even with a computer, though, the solution becomes quickly prohibitive due to the huge number of degrees of freedom.

Despite these premises, enormous progress has been obtained in the last twenty years, exploiting simplifications in the structure of quantum states that may occur in certain particular situations. To understand these simplifications we have to go through a more detailed study of the correlations that emerge in a quantum many-body state. In particular we have to borrow some instruments developed in the theory of quantum information.

In this chapter, we will go through the area laws: a change in the scaling of entanglement with the system size, that happens for ground states of local, gapped hamiltonians. We will first need to learn about partial trace, von Neumann entropy and Schmidt decomposition. Then, we will be able to talk about entanglement and how, for pure states, it is quantified by the entropy of entanglement. Finally, area laws will be introduced; they are of central importance in this work, because setting a limit to the amount of entanglement contained in the ground states enables us to identify methods which allow to efficiently represent them.

1.1 Reduced density matrix and partial trace

Before we begin, we would like to recall that, in quantum mechanics, the notion of pure vector state generalizes to the concept of density matrix.

Consider the situation where the system is in a mixed state: with probability p, the system is in the pure state $|\psi\rangle$ and, with probability $1-p$, it is in the pure state $|\phi\rangle$. If $O = O^\dagger$ is an observable, then its expectation value is straightforward to compute:

$$\langle O \rangle = p \langle \psi | O | \psi \rangle + (1-p) \langle \phi | O | \phi \rangle . \tag{1.1}$$

We define the *density matrix* (or density operator) ρ as:

$$\rho = p |\psi\rangle\langle\psi| + (1-p) |\phi\rangle\langle\phi| , \tag{1.2}$$

so that we can write:

$$\langle O \rangle = \mathrm{Tr}[\, \rho O \,] . \tag{1.3}$$

This compact way to write an expectation value holds for every possible mixed state; it suffices to generalize the density matrix:

$$\rho = \sum_i p_i |\psi_i\rangle\langle\psi_i| , \qquad \sum_i p_i = 1. \tag{1.4}$$

Moreover, every density matrix satisfies:

- $\rho \geq 0$ (positive-definite),
- $\mathrm{Tr}[\, \rho \,] = 1$ (normalization).

Now take a bipartite system AB and suppose that system A is associated with the m-dimensional Hilbert space $\mathcal{H}_A$ and system B is associated with the n-dimensional Hilbert space $\mathcal{H}_B$. We know that when two quantum systems are considered together, the Hilbert space where the global state lives is the tensor product of the single Hilbert spaces: $\mathcal{H}_{AB} = \mathcal{H}_A \otimes \mathcal{H}_B$.

Among the others, in $\mathcal{H}_{AB}$ we find states that cannot be expressed as the product of a state of $\mathcal{H}_A$ by a state of $\mathcal{H}_B$. As an example, take $|e_1\rangle_A \neq |e_2\rangle_A$ as two basis elements of $\mathcal{H}_A$ and then $|f_1\rangle_B \neq |f_2\rangle_B$ as two basis elements of $\mathcal{H}_B$. From the tensor product structure, $|e_1\rangle_A |f_1\rangle_B, |e_2\rangle_A |f_2\rangle_B \in \mathcal{H}_{AB}$ are two elements of a base of $\mathcal{H}_{AB}$ certainly different from each other. Then the superposition:

$$|\psi\rangle_{AB} = \frac{1}{\sqrt{2}} \left(|e_1\rangle_A |f_1\rangle_B + |e_2\rangle_A |f_2\rangle_B \right) , \tag{1.5}$$

is a legitimate state of the system which cannot be cast in a factorized form: $|\psi\rangle_{AB} \neq |\varphi\rangle_A \otimes |\chi\rangle_B$. It seems as if it is not possible to define a state for system A only, considered apart from B. This is not true, in fact, it is always possible to formally describe the information available to those who

can access system A. Indeed, in a laboratory we can measure system A alone and the probability distributions that we obtain from those measurements *define* the state of A. Suppose that we want to measure observable $\mathcal{O}_A$ and that the system is globally in a pure state $|\psi\rangle_{AB}$:

$$|\psi\rangle_{AB} = \sum_{i=1}^{m}\sum_{j=1}^{n} \psi_{ij}|i\rangle_A|j\rangle_B, \tag{1.6}$$

where $\{|i\rangle_A\}_{i=1,\dots,m}$ and $\{|j\rangle_B\}_{j=1,\dots,n}$ are basis for systems A and B respectively. Note that the coefficients of $|\psi\rangle_{AB}$ could be interpreted as the elements of an $m \times n$ matrix that we call Ψ: $(\Psi)_{ij} = \psi_{ij}$. We associate $\mathcal{O}_A$ with the operator $\mathcal{O}_A \otimes \mathbb{1}_B$, that is measurable on the entire system. The expectation value will be:

$$\langle \mathcal{O}_A \otimes \mathbb{1}_B \rangle = {}_{AB}\langle\psi|\mathcal{O}_A \otimes \mathbb{1}_B|\psi\rangle_{AB} = \sum_{i,j,k} \psi_{ij}\psi^*_{kj}{}_A\langle k|\mathcal{O}_A|i\rangle_A. \tag{1.7}$$

On the RHS, the degree of freedom of system B (the indices j's) are contracted and summed over. It is possible and reasonable to introduce a *state* for subsystem A that reproduces the same statistics of results:

$$\rho_A \coloneqq \sum_{i,j,k} \psi_{ij}\psi^*_{kj}|i\rangle_A\langle k| = \Psi\Psi^\dagger \tag{1.8}$$

$$\langle \mathcal{O}_A \rangle = \mathrm{Tr}[\,\rho_A \mathcal{O}_A\,]\,. \tag{1.9}$$

We call ρ_A, the *reduced density matrix* of A. This is indeed a well defined density matrix, because it is positive (as can be seen in terms of matrix Ψ) and with unitary trace. Likewise, we define the reduced density matrix for B as $\rho_B = (\Psi^\dagger\Psi)^t$. Both these definitions can be generalized to mixed states of system AB, but we will not need this here.

To compute the reduced density matrix it is convenient to introduce the formal operation of *partial trace*. The definition is given on an operatorial basis of AB:

$$\mathrm{Tr}_B\big[\,|a_i\rangle_A\langle a_j| \otimes |b_i\rangle_B\langle b_j|\,\big] \coloneqq |a_i\rangle_A\langle a_j|\,\mathrm{Tr}\big[\,|b_i\rangle_B\langle b_j|\,\big] = \big({}_B\langle b_j|b_i\rangle_B\big)\,|a_i\rangle_A\langle a_j|\,. \tag{1.10}$$

In this way, we write, for example, $\rho_A = \mathrm{Tr}_B[\,\rho_{AB}\,]$. The operation of partial trace throws away every information about system B including the possible correlations. In fact, in general $\rho_{AB} \neq \rho_A \otimes \rho_B$.

1.2 Shannon and von Neumann entropies

Shannon entropy is a key quantity in classical information theory. It describes directly the amount of missing information in a stochastic variable. In this

work, we are only interested in its structural properties and its connections with von Neumann entropy.

Consider a stochastic variable X such that $X = x_i$, $i = 1, \ldots, D$, with probability $p_i \geq 0$ and $\sum_{i=1}^{D} p_i = 1$. We define the *Shannon entropy* of X by:

$$H(X) \coloneqq -\sum_{i=1}^{D} p_i \log_2 p_i. \tag{1.11}$$

Remembering that this is an entropic quantity it is easy to guess for which distributions it will reach the maximum and the minimum value:

- if X has a probability distribution peaked only on one point, namely $p_i = \delta_{ik}$ with k fixed, then $H(X) = 0$ is the minimum value [1];
- if X has a flat probability distribution, namely $p_i = 1/D \; \forall i$, then $H(X)$ reaches the maximum value:

$$H(X) = -\frac{1}{D} \sum_{i=1}^{D} \log_2 \frac{1}{D} = \log_2 D. \tag{1.12}$$

In a quantum mechanical context, Shannon entropy translates to the *von Neumann entropy*:

$$S(\rho) = -\operatorname{Tr}[\, \rho \log_2 \rho \,]. \tag{1.13}$$

Initially, this was introduced to study the thermodynamical properties of quantum systems. The link between this quantity and Shannon entropy is through the eigenvalues of ρ (that exists, being ρ a positive operator) and the spectral theorem:

$$S(\rho) = -\sum_{i} \lambda_i \log_2 \lambda_i = H(\Lambda_\rho), \tag{1.14}$$

i.e. the von Neumann entropy of a density operator ρ coincides with the Shannon entropy of the distribution Λ_ρ of the eigenvalues of ρ. Here we used the von Neumann entropy defined with logarithm to base 2 instead of the natural logarithm, originally used for thermodynamic purposes. The difference is only in a proportionality constant and the reason why we take this definition is that the bit is the natural measure for information.

Like Shannon entropy, von Neumann entropy has a maximum and minimum value too:

[1] We defined $0 \log_2 0 \coloneqq 0$.

- pure states always have zero entropy, this is easy to see because $\rho^{(\min)} = |\psi\rangle\langle\psi|$ is a projector on a one-dimensional subspace, so the spectrum of ρ will have only one eigenvalue equal to one and all the others are zeros; this is the peaked distribution discussed above, the minimum value;
- the maximum value is reached for completely mixed states $\rho^{(\max)} = \mathbb{1}/D$, where D is the dimension of the Hilbert space; the spectrum here is read directly: flat with all eigenvalues equal to $1/D$, the same distribution that maximizes $H(X)$, $S(\rho^{(\max)}) = \log_2 D$.

This entropy is a measure of the grade of mixture of the state.

Other important properties of von Neumann entropy to be mentioned here are:

- the entropy does not change under unitary evolution (or change of base), $S(U\rho U^\dagger) = S(\rho)$. This can be derived from the cyclic rule of the trace and the fact that every function f, that is expandable as a power series, satisfies $f(UAU^\dagger) = Uf(A)U^\dagger$, as the logarithm does;
- the entropy of a tensor product is the sum of the entropies of the subsystems: $S(\rho_1 \otimes \rho_2) = S(\rho_1) + S(\rho_2)$.

1.3 Schmidt decomposition

The Schmidt decomposition is a very useful form in which we can express pure states of bipartite systems. It is a corollary of the singular value decomposition (SVD) applied in a quantum mechanical context. The SVD is described in some details in appendix A; the Schmidt decomposition inherits many of its properties.

Theorem 1.1 (Schmidt decomposition)**.** *Let $|\psi\rangle_{AB} \in \mathcal{H}_{AB}$ be a pure bipartite state as in* (1.6)*, then there exists a basis $\{|e_k\rangle_A\}_{k=1,\dots,m} \subset \mathcal{H}_A$, a basis $\{|f_k\rangle_B\}_{k=1,\dots,n} \subset \mathcal{H}_B$ and a set of non-negative real numbers $\{\sigma_k\}_{k=1,\dots,p}$ such that we can write:*

$$|\psi\rangle_{AB} = \sum_{k=1}^{p} \sigma_k |e_k\rangle_A |f_k\rangle_B \tag{1.15}$$

Proof. Consider the coefficients ψ_{ij} of $|\psi\rangle_{AB}$ that form the matrix Ψ of dimensions $m \times n$ as described in section 1.1. Now apply the singular value decomposition in reduced form[2] to the coefficients:

$$\psi_{ij} = \sum_{k=1}^{p} u_{ik}\sigma_k v^\dagger_{kj}; \tag{1.16}$$

[2]See again appendix A.

and substitute this in (1.6):

$$|\psi\rangle_{AB} = \sum_{i=1}^{m}\sum_{j=1}^{n}\sum_{k=1}^{p} u_{ik}\sigma_k v_{kj}^{\dagger}|i\rangle_A|j\rangle_B \tag{1.17}$$

$$= \sum_{k=1}^{p}\sigma_k \left(\sum_{i=1}^{m} u_{ik}|i\rangle_A\right)\left(\sum_{j=1}^{n} v_{kj}^{\dagger}|j\rangle_B\right). \tag{1.18}$$

Now we define the first p vectors of each basis as:

$$|e_k\rangle_A \coloneqq \sum_{i=1}^{m} u_{ik}|i\rangle_A, \quad |f_k\rangle_B \coloneqq \sum_{j=1}^{n} v_{kj}^{\dagger}|j\rangle_B, \quad k = 1,\ldots,p. \tag{1.19}$$

The orthonormality derives from coefficients u_{ik} and $v_{kj}^{\dagger}$ that form isometric matrices (that act on basis elements). Then it is possible to complete the basis with the missing vectors however we want (e.g. with a Gram-Schmidt process). The positivity of reals σ_k follows from the SVD. □

The number of coefficients $\{\sigma_k\}_k$ different from zero is called *Schmidt rank*, indicated with letter r. Note that the rank is always limited by the dimensions of the Hilbert spaces: $r \leq \min(m,n)$ as descends from SVD.

Some of the important derivations that we obtain from this decomposition are those for the reduced matrices of a pure bipartite state. For example:

$$\rho_A = \mathrm{Tr}_B\big[\,|\psi\rangle_{AB}\langle\psi|\,\big] = \sum_k \sigma_k^2 |e_k\rangle_A\langle e_k|\,, \tag{1.20}$$

i.e. the $\{|e_k\rangle_A\}_k$ are the eigenvectors of ρ_A (thanks to their orthonormality) and the eigenvalues of ρ_A are the squares of the Schmidt coefficients (the singular values of Ψ). The same calculations for ρ_B show:

$$\rho_B = \mathrm{Tr}_A\big[\,|\psi\rangle_{AB}\langle\psi|\,\big] = \sum_k \sigma_k^2 |f_k\rangle_B\langle f_k|\,, \tag{1.21}$$

where the eigenvectors are $\{|f_k\rangle_B\}_k$, but the eigenvalues are still the squares of the Schmidt coefficients. This is a very important result: the spectra of the two reductions of a pure bipartite state are the same, $\sigma(\rho_A) = \sigma(\rho_B)$.

We want to stress the importance of the orthonormality within the two vector sets $\{|e_k\rangle_A\}_k$ and $\{|f_k\rangle_B\}_k$: if we could express $|\psi\rangle_{AB}$ as in (1.15), but without the orthonormality, then we would not obtain the same properties for the reduced density matrices.

1.4 Entanglement and measures

The concept of entanglement emerges naturally from the tensor product structures described in section 1.1 and it is connected to the impossibility to write certain global states as product states. Many see the entanglement as one of the most stricking feature of quantum mechanics, because of its highly non-classical features. Another interesting aspect of entanglement is its (relatively long) history and how the gradual study of this curious property lead to a more mature quantum mechanics.

In what follows we briefly point out the fundamental milestones of entanglement theory. In 1932, von Neumann had already laid a solid mathematical basis for nonrelativistic quantum mechanics [18], but it was not until 1935 that the strange behaviour of certain quantum states was recognised. Einstein, Podolsky and Rosen (EPR), quite ironically, tried to use an entangled state to prove that quantum theory was not complete [19]. In the same year, Schrödinger, after reading the EPR paper, investigated the state they used and discovered the two main manifestations of entanglement [20]:

- when the entangled subsystems are measured, the statistical outcomes are correlated;
- the state could present greater order globally rather than in the single components.

In 1964, Bell was able to prove that the correlations in the outcomes cannot be described classically. More precisely, he proved that every (local) hidden variable model (LHVM) must obey particular inequalities and that quantum mechanics violates these inequalities. A LHVM is the most general classical theory that could be underlying quantum mechanics. It is based on the assumptions of *realism*—every system has well defined properties that exist prior to and are independent from measurements—and *locality*—two space-like separated events are independent. The quantum states that violate Bell inequalities are all entangled, thus they cannot be described classically. This was a great theoretical jump that shed new light and drew more attention to a field that was only the subject of philosophical debate. Still, the situation did not change too much because these "stronger" correlations exhibited by entanglement, even at space-like separation, do not allow to exchange information. The point of view definitely changed in 1991, when Ekert devised a protocol for quantum cryptography using entangled states and the Bell inequalities [21]. Exploiting entanglement, it was possible to securely share strings of random bits, a task not feasible with classical means. The subsequent discoveries of dense coding (1992, [22]) and quantum teleportation (1993, [23]) confirmed the status of resource of the entanglement.

As with every resource, one must be able to identify, detect, measure and manipulate entanglement. Here we are interested in a very simple measure for pure entangled states, because the complexity of the simulation of a quantum state increases with the entanglement content of the state. Having even a simple measure of entanglement will help us understand how efficiently we will be able to simulate a particular state.

We will limit ourselves to results of entanglement theory only for nonrelativistic, bipartite systems in pure states. The reason is twofold: on the one hand, for the purpose of this work it is all that we need, on the other hand, the same theories that treat multipartite or relativistic entanglement are far from complete. A general exposition at this point would be just a digression and outside the scope of this presentation[3].

1.4.1 Definition

Entanglement is a structural property of quantum states of compound systems. Compound systems can be shared among different laboratories, where scientists can perform quantum operations coherently on the part of the system *local* to them. These actions need not be independent, in fact, communications from one party to another can easily be established through classical means like the telephone or a set of computers over a network. This set of operations is called *Local Operations and Classical Communications* (LOCC).

Starting from a completely uncorrelated state $\rho = \rho_1 \otimes \rho_2 \otimes \cdots \otimes \rho_N$, we wonder what are the states that can be generated applying only LOCC operations. This is the set of separable states [27]:

Definition 1.2 (Separable states)**.** *We say that ρ is separable if there exists a probability distribution $\{p_i\}_i$ ($0 \leq p_i \leq 1$, $\forall i$, $\sum_i p_i = 1$) and a set of states $\{\rho_1^{(i)}, \ldots, \rho_N^{(i)}\}_i$ for every subsystem and for every event of the probability distribution, such that we can write:*

$$\rho = \sum_i p_i \, \rho_1^{(i)} \otimes \cdots \otimes \rho_N^{(i)}. \tag{1.22}$$

These states include every possible classical state (even those from classical probability theories) so they can describe every classical correlation. Every state that is not separable is called *entangled* and every correlation

[3]For further details, we refer the reader to the resources that were the starting point of this discussion: [24] finely explained classical results, [25] a complete review and [26] with special considerations for entanglement measures.

that cannot be simulated by a separable state is called a *quantum correlation*[4]. This is a broader definition of quantum correlations than that provided for long time by violation of Bell inequalities; in fact, for any Bell inequality there are entangled state that do not violate it and so they admit a LHV model, in other words, their correlations are still classical. Our choice here is justified by a recent discovery of Masanes [28] that every non-separable state is useful for some quantum process.

1.4.2 Entanglement measures

Looking at entanglement as a resource is what lets us quantify it. LOCC operations cannot create entanglement, thus entanglement is the resource that allows tasks not possible with LOCC alone. We need a notion that lets us distinguish which state is more entangled between two entangled states. The answer is simple in principle: ρ is more or equally entangled than σ, if ρ can be transformed into σ with LOCC. Often it is not possible to transform deterministically a state into another with LOCC, so a more rigorous approach is to transform a lot of states and to look at the rates of successful transformations. Summarizing, we give the definition of measure of entanglement for bipartite systems.

Definition 1.3 (Bipartite entanglement measure)**.** *A bipartite entanglement measure is a state function with non-negative real values:* $E : \rho \mapsto E(\rho) \in [0, +\infty)$*. It must respect the basic properties of entanglement:*

- *separable state have no entanglement:* ρ *separable* $\implies E(\rho) = 0$ *;*
- *LOCC transformations do not increase entanglement on average: if* $\rho \to \sigma$ *through LOCC, then* $E(\sigma) \leq E(\rho)$*.*

The second property implies that entanglement does not change under local unitaries, because they are LOCC transformations that can be inverted, so:

$$E(U_A \otimes V_B \, \rho \, U_A^\dagger \otimes V_B^\dagger) = E(\rho). \tag{1.23}$$

In the case of bipartite systems only, there exist states that maximize entanglement. These are pure states from which every other state can be obtained (on average) with LOCC transformations. In a bipartite system formed by two D-dimensional parties, these states are those unitary equivalent to:

$$|\chi_D\rangle_{AB} = \frac{1}{\sqrt{D}} \sum_{j=1}^{D} |j\rangle_A \otimes |j\rangle_B. \tag{1.24}$$

[4]By quantum correlations we do not mean non-classical correlations.

It is reasonable to require a normalization of the entanglement measure such that $E(|\chi_D\rangle_{AB}\langle\chi_D|) = \log_2 D$.

1.4.3 Entropy of entanglement

Consider the function $E_E \coloneqq S \circ \mathrm{Tr}_B$, where S is the entropy of von Neumann. When applied to pure bipartite states only, this function defines a measure for entanglement that is called *entropy of entanglement*. In details, from a pure state $\rho_{AB} = |\psi\rangle_{AB}\langle\psi|$ it is extracted the reduced density matrix ρ_A and then the von Neumann entropy of this matrix is calculated: $E_E(\rho_{AB}) = S(\mathrm{Tr}_B[\,\rho_{AB}\,]) = S(\rho_A)$.

Many properties of this measure derive directly from the Schmidt decomposition. In particular there is no difference in taking the partial trace over part B or part A: the spectra of the reduced density matrices ρ_A and ρ_B from a pure state are identical, thus $S(\rho_A) = S(\rho_B)$.

In simple terms, the justification of this measure derives from the fact that the more entangled the state is, the less we know about its constituents. If $|\psi\rangle_{AB}$ is in a separable state, then the Schmidt rank is 1 and the state can be written as $|a\rangle_A|b\rangle_B$. The reduced density matrix is still a pure state: $\rho_A = |a\rangle_A\langle a|$ so we have complete knowledge of subsystem A (and of course of subsystem B too). Therefore we see that the entropy of entanglement is zero for a separable state: $E_E(\rho_{AB}) = S(|a\rangle_A\langle a|) = 0$, which means no entanglement.

On the other hand, when the Schmidt rank of $|\psi\rangle_{AB}$ is greater than 1, the state is non-separable and the value of the entropy of entanglement will depend on the spectrum of the reduced density matrix. A flatter distribution of eigenvalues means less knowledge and, thus, more entanglement. The maximum is reached, correctly, by those completely flat spectra resulting from Bell-like states as in (1.24).

1.5 Area laws

When we say that a quantity follows an *area law*, we mean that it does not grows with the size of the system but with the size of the boundary of the system. Here we give a slightly more precise, but still fuzzy definition:

Definition 1.4 (Area and volume laws)**.** *Take a system A (that could be a part of a larger entity) and suppose we can assign a size $\mathcal{S}(A)$ to it; in turns, this means that there exists a characteristic length of A, called $\mathcal{L}$, such that:*

$$\mathcal{S}(A) \propto \mathcal{L}^{\mathcal{D}}, \tag{1.25}$$

where $\mathcal{D}$ is the dimensionality of the system. We say that a property P relative to A follows an area law if it is proportional to the area of the system:

$$P \propto \mathcal{A}(A) \propto \mathcal{L}^{\mathcal{D}-1}. \tag{1.26}$$

Instead, when the property scales directly with the system size, we say that it follows a volume law*.*

The exact nature of the characteristic length and of the size and surface of the system is very problem-dependent. Needless to say, the words "area laws" and "volume laws" come from the three-dimensional case, where boundary size and system size do correspond to the area and volume of the system. An important review of the research in area laws is [12] and the next considerations are taken from it.

1.5.1 History

Area laws are rare in physics and the only known quantities that present them are entropies. The first appearance of an area law was with the Bekenstein-Hawking black hole (thermodynamic) entropy [29, 30]. This triggered the study of geometric entropies, i.e. entropy of subregions (related to the entropy of entanglement), in quantum field theory [31, 32] and conformal field theory [33, 34, 35, 36, 37], since they showed very similar area-like scaling.

In the early 2000s, a series of articles began to investigate correlations in quantum many-body systems borrowing concepts from quantum information [38, 39, 40, 11]. They found area laws for the entropy of entanglement too. Shortly after, Hastings proved that a general area law exists for ground states of many-body systems with a local, gapped hamiltonian in 1D [41]. We want to make clear that an area law in 1D means that the entropy does not grow and it is limited by a finite constant.

For higher dimensions, at the time we are writing, there are only partial results and a general area law is still missing[5]. Anyway, it is a strongly supported conjecture.

1.5.2 Motivations

In what follows, we try to give some of the motivations that drive the interest in the study of area laws, especially in many-body theories. We list them from the least important to the most important *for this work*; they are all very important in their own rights.

[5]The reasons reside in the analytical difficulties that appear when the translational symmetry is broken considering only a subpart of the system.

- The presence of area laws in both relativistic and nonrelativistic theories suggests that high-energy physics questions with area-like behaviour, such as the already cited black hole entropy, the AdS/CFT correspondence [42, 43] and the holographic principle [44, 45], could have the same microscopic explanations of many-body systems and they could be deeply connected with the locality of interactions.

- Many-body systems are often organized as lattices and lattices are sometimes used to discretize quantum field theories [46]. This is why area laws could also aid the study of correlations in continuous theories.

- Area laws are useful to investigate the phenomenon of quantum phase transition and the distribution of correlations in many-body ground states, since it has been shown that entanglement exhibits a different behaviour near a point of criticality [47].

- Finally, the most important reason, for us, to study area laws is their connection with simulability. Indeed, they are the main reason of the success of the Density Matrix Renormalization Group described in section 3.1. They signal a small amount of entanglement in ground states of quantum many-body systems with local interactions. It is easy to understand intuitively that this means a small degree of complexity and that an easier representation for those ground states is possible. This intuition has been formalized in [48], which says that if the scaling of entanglement in a many-body state is suitably limited, then that state can be approximated efficiently by a Matrix Product State (see chapter 2).

1.5.3 Lattices and hamiltonians

To understand what an area law for the entropy of entanglement means to us, we have to enter more in the details of the problems studied.

We treat only *locally finite* many-body systems on a lattice. This means that for every site of the lattice there is a quantum system described by a finite-dimensional Hilbert space. There is no need, at this point, for the lattice to have good ordering, so we adopt the broad definition of general lattices. A *general lattice* is a graph $G = (L, E)$, where L is a set of points in space and $E \subset L \times L$ is the *edges set*, containing pairs of those points. A point in L could be $p = (x_1, \ldots, x_{\mathcal{D}}) \in L$, where $\mathcal{D}$ is the dimensionality of the system, and a pair of E is $(p_i, p_j) \in E$, such that $p_i, p_j \in L$. The presence of a pair in E indicates whether or not the two points of the pair are *nearest*

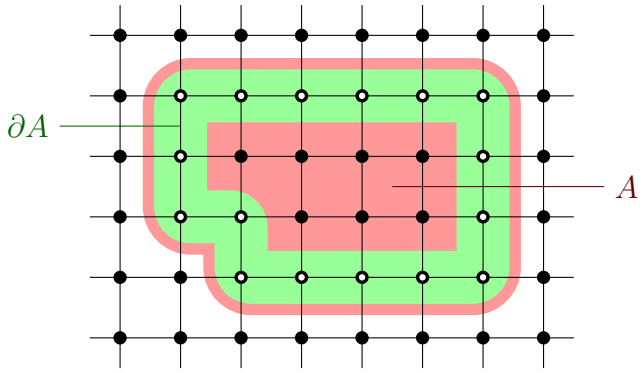


Figure 1.1: Example of regular square lattice with bulk and boundary of a subregion A.

neighbor. We define recursively a *path* that joins $p_i, p_j \in L$ as a subset of E, $P_{ij} \subset E$:

$$P_{ij} = \begin{cases} \{(p_i, p_j)\} & \text{if } (p_i, p_j) \in E \\ \{(p_i, p_k)\} \cup P_{kj} & \text{if } (p_i, p_k) \in E \end{cases}. \tag{1.27}$$

We indicate the *size* of a set with $|\cdot|$, thus, if L is finite, the number of sites is $N = |L|$. It is easy to define a *distance* between two sites as the length of the shortest path between them:

$$\mathrm{dist}(i, j) = \min_{P_{ij}} |P_{ij}|\,; \tag{1.28}$$

and it follows immediately that two sites are nearest neighbor if and only if the distance between them is 1. Finally, if we identify a subset of the lattice by $X \subset L$ and its complement by $L \setminus X$, then the *boundary* of X is the set of all points of X with a nearest neighbor outside of it (see for instance figure 1.1):

$$\partial X = \{p_i \in X \,|\, \exists\, p_j \in L \setminus X \text{ such that } \mathrm{dist}(i, j) = 1\}. \tag{1.29}$$

We assume that all the local Hilbert spaces $\mathcal{H}_i$ have dimension $\dim \mathcal{H}_i = d$. The total Hilbert space will be the tensor product:

$$\mathcal{H} = \bigotimes_{p_i \in L} \mathcal{H}_i, \qquad \dim \mathcal{H} = d^N. \tag{1.30}$$

Some words must be spent on the nature of the interactions too. The hamiltonians are *local*, meaning that each term has support on a *compact* subset X of L:

$$H = \sum_{X \subset L} H_X. \tag{1.31}$$

This has strong implications on the definition of the dimensionality of the problem itself. Indeed, a problem with nearest-neighbor interactions on a regular square lattice, ($\mathcal{D} = 2$), with ℓ sites per side, could be easily restated on a one-dimensional chain, ($\mathcal{D} = 1$), if we allow arbitrary long interactions. For example, take two nearest neighbors, p_i and p_j, in the square lattice; when we unwind the system as a long chain, the two sites will be at a maximum distance of ℓ, the side of the square. If the hamiltonian contains proper interactions at distance ℓ we have reduced the bi-dimensional problem to a one-dimensional problem. Imposing a locality constraint on the interactions, we avoid these situations. Systems with different dimensionalities belong to different classes of complexity and this is testified exactly by the area laws.

1.5.4 The typical case: volume laws

Let us return to the entropy of entanglement defined for the bipartite system AB with Hilbert space $\mathcal{H}_{AB} = \mathcal{H}_A \otimes \mathcal{H}_B$. If $\dim \mathcal{H}_A = m$ and $\dim \mathcal{H}_B = n$, then $\dim \mathcal{H}_{AB} = mn$. Suppose, without loss of generality, that $m \leq n$, then we know that there exist states as entangled as (1.24), with $D = m$:

$$E_E\left(|\chi_m\rangle_{AB}\langle\chi_m|\right) = \log_2 m. \tag{1.32}$$

A natural question that may arise is whether these maximally entangled states are frequent or not. In other words, if we randomly choose a pure state, we would like to know how much it will be entangled. Page answered in the limit of large m and n [49], that is the case we are interested in:

Theorem 1.5 (Page's theorem)**.** *Consider system AB as above. The average $\langle\,\cdot\,\rangle_{m,n}$ of the entropy of entanglement is weighted with respect to the unitarily invariant Haar measure*[6] *of the space of unit vectors $|\psi\rangle_{AB}$ in $\mathbb{C}^{mn}$. For large m and n, it holds:*

$$\left\langle E_E\left(|\psi\rangle_{AB}\langle\psi|\right)\right\rangle_{m,n} \approx \log_2 m - \frac{m}{2n} \qquad \text{for } n \geq m \gg 1. \tag{1.33}$$

This is impressive: in the thermodynamic limit, $n \to \infty$, the average entanglement of a random state is always maximal.

Contextualize this result in the lattice framework: in every site there is a d-dimensional Hilbert space that describes the local physics. If we divide L

[6]This is the formally correct name of the measure proportional to the volume of the $(2mn-1)$-dimensional hypersphere that we need since the states live in $\mathbb{C}^{mn}$ that has mn complex dimensions.

in the two subsets $A \subset L$ and $B = L \setminus A$, we will have $m = \dim \mathcal{H}_A = d^{|A|}$, $n = d^{|B|} = d^{N-|A|}$, which means that equation (1.33) becomes:

$$\langle S(\rho_A) \rangle \approx |A| \log_2 d - \frac{1}{2d^{N-2|A|}}. \tag{1.34}$$

As $N \to \infty$, this approximate a volume law. The average entropy of entanglement of a random many-body state is always maximal and proportional to the *size* of the block considered.

1.5.5 The rare case: area laws

Looking at how, almost always, a random state follows a volume law, then we can safely assume that states with an area law are extremely rare. In the context of lattices, an area law implies that the entropy grows like the size of the boundary of the subregion considered:

$$S(\rho_A) \approx |\partial A| \,. \tag{1.35}$$

As already mentioned, for $\mathcal{D} = 1$ there exists a very important theorem proved by Hastings [41] which affirms that the entropy of entanglement for every ground state of a local, gapped hamiltonian is bounded by a constant. This is exactly an area law in one dimension, because the boundary of every contiguous region contains at most two sites and does not grow.

For Hastings' theorem, we can consider all interactions to be between nearest neighbors, because the local dimension d is arbitrary[7]:

Theorem 1.6 (Hastings' theorem)**.** *Consider a one-dimensional system on a lattice L governed by a hamiltonian H with only nearest-neighbor interactions and every interaction with a finite strength:*

$$H = \sum_{p_i \in L} H_{i,i+1}, \qquad \|H_{i,i+1}\| \leq J, \text{ for some } J > 0. \tag{1.36}$$

Suppose H has a gap between the ground state energy (ε_0) and the first excited state energy (ε_1) different from zero, $g = \varepsilon_1 - \varepsilon_0 > 0$. Then there exists a constant $S_{max} > 0$ such that, for every subregion $A \subset L$:

$$S(\rho_A) \leq S_{max}. \tag{1.37}$$

[7]For example, if the longest range for our interactions is ℓ, we can regroup every ℓ contiguous sites in a single local component of dimension $d' = d^\ell$ and then we would return to nearest-neighbor ranges only.

It is very important to understand the incredible luck that this is. Before, if we wanted to identify the ground state of a many-body system, we had to look for it in a Hilbert space with an exponential number of dimensions. Now that we know about this area law, we can limit to a much narrower subspace because ground states are *very* uncommon. The question shifts to how we can parametrize this small, special subspace and the answer is: with Matrix Product States (see chapter 2).

1.5.6 Critical systems

Hastings' theorem excludes the case when the gap is null, $g = 0$, namely when the system is critical. At criticality, the correlation length diverges and the system is scale invariant. These situations are well described by conformal field theory which studies the *conformal group*: to the Poincaré group (translations and rotations), all the scaling transformations are added. Conformal field theory finds small deviations from the exact area law for one-dimensional ground states, in the form of a logarithmic growth with the subregion size:

$$S(\rho_A) \approx \log_2(|A|). \tag{1.38}$$

Chapter 2

Matrix Product States

In this chapter, we introduce the theory of Matrix Product States (MPS); they are a new way to write many-body quantum states as strings of matrix products. Currently, they are the best parametrization for states presenting an area law in one dimension. Because of what we have seen in section 1.5.5, they are the optimal class of vectors where to look for the ground state of a local, gapped hamiltonian.

We will accumulate a lot of information about MPSs, almost all of which will be used to build the new algorithm in section 3.2; let us go through it step by step. Initially, we present MPSs in form of ansatz, trying to elucidate their meaning and, afterwards, we give some examples to make them easier to understand. The choice of the matrices that define an MPS is not unique and there is a true gauge freedom that determine the "normalization" that these matrices have. The comprehension of the mechanism that changes such normalization is vital for some operations like expectation values of simple operators and compression.

At this point, we will be ready to better justify the origin of MPSs and we will give two approaches. The first one, the valence bond derivation, starts from a very interesting abstract setting and it clearly shows how to obtain a vector state with area-like scaling of entanglement. The second one, instead, involves a very common tool in this work, the singular value decomposition, and the result will be that every state can be written in MPS form allowing arbitrary dimensions for the matrices. In this case, the area law is introduced imposing, by hand, a cutoff for the matrices.

In the second part of this chapter, we begin exploring the advanced algorithms. We will do it with the help of a very simple graphical notation for the objects we play around with. These advanced algorithms are the computation of overlaps and expectation values with transfer operators, the MPS sum and the MPS compression. Indeed, we will see that the sum increases

the size of an MPS and, thus, we will be forced to introduce a procedure to reduce this size. We will show two ways to compress an MPS: with the singular value decomposition of appendix A or with a variational update of the matrices.

In the last part, then, we generalize the MPS formalism for operators, called Matrix Product Operators (MPOs), which we need to guarantee that, after an application of the operator, the MPS form is preserved.

Almost everything we will say in this chapter comes from the review of Schollwöck [15], which is far more complete and, thus, we point to it for any doubt or further interest.

2.1 MPS ansatz

Before giving the definition of matrix product states, we review the kind of problems we are studying and we specify the conventions used for one-dimensional lattices. Indeed, matrix product states have slightly different definitions depending on the structure of the lattice.

We are focusing on many-body systems with the constituents arranged on a lattice. In every site of the lattice, the local quantum system lives in a finite Hilbert space and the total Hilbert space is given by the tensor product of these local spaces (see equation (1.30)). We are only interested in those problems where the local Hilbert space dimension is fixed to d. Calling N the number of sites, we can write a generic vector state for these kinds of systems as:

$$|\psi\rangle = \sum_{i_1,\dots,i_N=1}^{d} c_{i_1\cdots i_N} |i_1 \cdots i_N\rangle , \qquad \langle\psi|\psi\rangle = 1, \tag{2.1}$$

where $c_{i_1\cdots i_N}$ are d^N complex coefficients, and $|i_1 \cdots i_N\rangle$ are d^N basis elements of the total Hilbert space.

The knowledge of the ground state is of fundamental importance for the study of these systems as it is usually the first information about the spectrum that we can find. Often, we have no clue about the ground state and the only remaining possibility is to look for it in the enormous class of states of the form (2.1), choosing the one with the lowest energy.

In section 1.5.5, we found that ground states of gapped, local hamiltonians in 1D have bounded entropy of entanglement for every possible bipartition of the system. This is a major simplification for the structure of ground states and there is hope to find a smaller class of states that may always contain the ground state we are looking for.

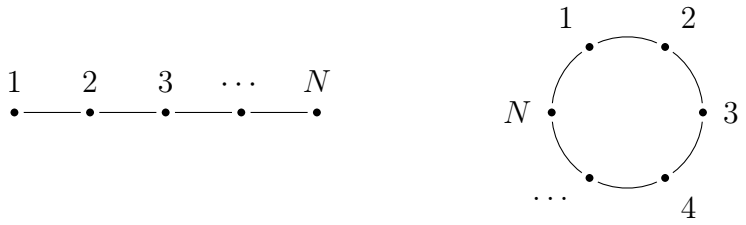


Figure 2.1: On the left: a 1D lattice with open boundary conditions (OBC). On the right: a 1D lattice with periodic boundary conditions (PBC).

From now on, we will adopt this setting: a system on a regular lattice of dimensionality $\mathcal{D} = 1$, that we also call a *chain*, with only local, gapped hamiltonians. We remind that local hamiltonians have interactions with finite range and "gapped" means there is a non-zero energy difference between ground state energy and first excited state energy.

Regular one-dimensional lattices come in two forms: with *periodic boundary conditions* (PBC) or *open boundary conditions* (OBC). In the first case, the first and the last site of the lattice are nearest neighbors, in the second case, they are not (see figure 2.1). To be more precise, using the notation of section 1.5.3, we denote the set of points L of the lattice just as a subset of the natural numbers, $L = \{1, 2, \ldots, N\}$: since the lattice is regular, we implicitly substituted the positions of the sites in L by numbers. If the chain is drawn horizontally, the sites are numbered in increasing order from left to right; this is just a convention since there is no physical meaning for "left" or "right" in this situation. For both OBC and PBC, the edges set E contains all the pair $(s, s+1) \in E$, with $s = 1, \ldots, N-1$. The difference is that PBC lattices include also $(N, 1) \in E$, while OBC lattices do not. We will often refer to the edges of the lattice as *bonds*.

A *Matrix Product State* (MPS) is a new *parametrization* for a generic vector state like (2.1). MPSs substitute the knowledge of the tensor coefficients $c_{i_1 \cdots i_N}$, with the knowledge of a set of matrices $\{M_i^{[s]}\}_{i=1,\ldots,d}^{s=1,\ldots,N}$. In particular, if the system has PBC, we can generate from these matrices the state:

$$|\psi\rangle = \sum_{i_1,\ldots,i_N=1}^{d} \mathrm{Tr}\Big[M_{i_1}^{[1]} \cdots M_{i_N}^{[N]} \Big] \, |i_1 \cdots i_N\rangle \,. \tag{2.2}$$

Otherwise, if the system has OBC, we write the state as:

$$|\psi\rangle = \sum_{i_1,\ldots,i_N=1}^{d} M_{i_1}^{[1]} \cdots M_{i_N}^{[N]} \, |i_1 \cdots i_N\rangle \,, \tag{2.3}$$

where the first and the last matrices are respectively row and column vectors. In this way we obtain a scalar coefficient for every base element $|i_1 \cdots i_N\rangle$.

A few comments on these expressions: MPSs assign d matrices for every site. The shape of the matrices is quite free and the optimal sizes are very problem-dependent. Anyway there are some simple rules to follow:

- every matrix on the same site has the same number of rows and columns;
- matrices of adjacent sites must have columns and rows that respect matrix multiplication rules;
- (for OBC only) matrices on the first and last site are row and column vectors respectively.

From these points, we infer the following dimensions:

$$M_i^{[s]} : D_{s-1} \times D_s \qquad \forall i, \tag{2.4}$$

and in case of OBC we also have the constraint for first and last matrices:

$$M_i^{[1]} : 1 \times D_1 \qquad M_i^{[N]} : D_{N-1} \times 1. \tag{2.5}$$

The number of values $\{D_s\}_s$ is equal to the number of bonds, so N for PBC and $N-1$ for OBC. For an MPS, the connection between two matrices of adjacent sites is also named *bond link* and this is why the values D_s are called *bond link dimensions*; they are, indeed, a property of the bond. Every bond or bond link is numbered with the same rule as for the D_s: the number of the site on their left.

It is possible to use the bond link dimensions to divide the set of all MPSs into *classes*. We group together all MPSs with the same maximum bond link dimension D. We also use this dimension to label the MPS class. Thus, if an MPS is of class D, it means $D_s \leq D$, $\forall s$ (and there also exists t such that $D_t = D$).

2.2 Examples

The idea of MPS seems a bit abstract at this point; we show some examples of many-body states written in MPS language to make that idea more concrete. We will not be very precise with the normalization of these states as it is not immediate to define what normalization means for an MPS, we will address the issue in section 2.3.

1. Product states: they are a special case of MPS with all the matrices substituted by numbers. Let $|\psi\rangle = |\phi^1\rangle|\phi^2\rangle\cdots|\phi^N\rangle$ and $|\phi^s\rangle = \sum_{i=1}^{d} c_i^s\,|i\rangle$. Then, all it takes is to put:

$$M_i^{[s]} = c_i^s, \quad \forall\, i, s. \tag{2.6}$$

Among the vectors that can be represented in this way, there are all the computational basis elements (where the c's become Kronecker deltas).

2. GHZ state [50]: $|\mathrm{GHZ}\rangle = |00\cdots0\rangle + |11\cdots1\rangle$, for $d = 2$. Here we increase the difficulty a bit. Since this state is the sum of two basis elements, we will see in section 2.8 that we can expect the MPS matrices to be no more than 2×2. Not only is this true, but also the matrices are site-invariant. An intuitive way to find the desired matrices is this: every time a matrix for index $i = 0$ ($i = 1$) multiplies an adjacent matrix for index $i = 1$ ($i = 0$) the result must be zero. On the other hand, when two matrices for the same index are multiplied together the result is that same matrix (a projector or idempotent matrix). It is easy to verify that these are the required matrices:

$$M_{i=0}^{[s]} = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}; \qquad M_{i=1}^{[s]} = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix}, \tag{2.7}$$

with the first and last site matrices (in OBC case):

$$\begin{aligned} M_0^{[1]} &= \begin{pmatrix} 1 & 0 \end{pmatrix} \quad M_1^{[1]} = \begin{pmatrix} 0 & 1 \end{pmatrix} \\ M_0^{[N]} &= \begin{pmatrix} 1 \\ 0 \end{pmatrix} \quad M_1^{[N]} = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \end{aligned} \tag{2.8}$$

3. W state: $|W\rangle = |10\cdots0\rangle + |010\cdots0\rangle + \cdots + |0\cdots01\rangle$, $d = 2$. This is a sum of N basis elements, in this case the MPS matrices should be no more than $N \times N$. In fact, they are still 2×2, and again site-invariant. Here only the matrix for index $i = 0$ is a projector and when it is multiplied to a matrix for index $i = 1$, the result must be the latter. Two matrices for index $i = 1$ multiplied together give zero (nilpotent matrix). What happens is that, when we fix the basis element $|j_1\cdots j_N\rangle$, the series of matrix products in front of it determines whether that element survives or not. Reading from left to right the kets, we keep accepting $|0\rangle$'s maintaining the same matrix, until we find a $|1\rangle$, and the matrix changes. This remember us that we already found $|1\rangle$, then if we find another one the basis element must die, with coefficient zero. Otherwise, if there are only $|0\rangle$'s in what follows, the

basis element is good and we let it live with coefficient 1. The following matrices implement what just said:

$$M_0^{[s]} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}; \qquad M_1^{[s]} = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}, \tag{2.9}$$

with the first and last site matrices:

$$\begin{array}{cc} M_0^{[1]} = \begin{pmatrix} 1 & 0 \end{pmatrix} & M_1^{[1]} = \begin{pmatrix} 0 & 1 \end{pmatrix} \\ M_0^{[N]} = \begin{pmatrix} 0 \\ 1 \end{pmatrix} & M_1^{[N]} = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \end{array} \tag{2.10}$$

We will see again this sort of "matrix computation" in section 2.13.

4. AKLT state: named after its discoverers [51]. It is historically important for three reasons: firstly, it was the first example of non-trivial ground state that could be expressed in MPS form with matrices that do not grow with the system size; secondly, the way it was found inspired the valence bond derivation delineated in section 2.4; thirdly, it triggered an interest in the class of *finitely correlated states* (FCS, [52]), a mathematical physics approach to MPSs parallel and independent from DMRG. The AKLT state is the ground state of the AKLT hamiltonian:

$$H = \sum_i \left(\mathbf{S}_i \cdot \mathbf{S}_{i+1} + \frac{1}{3} (\mathbf{S}_i \cdot \mathbf{S}_{i+1})^2 \right). \tag{2.11}$$

The $\mathbf{S}_i$ are the spin-1 generators, thus $d = 3$. We do not give its full derivation here, since it is a special case of the procedure in section 2.4; see, for example, [15] for further details. The dimension of the bond link is still very small, $D = 2$, and the matrices are the same for every site (it is a PBC MPS):

$$M_0^{[s]} = \begin{pmatrix} 0 & \sqrt{\frac{2}{3}} \\ 0 & 0 \end{pmatrix}, \quad M_1^{[s]} = \begin{pmatrix} -\frac{1}{\sqrt{3}} & 0 \\ 0 & \frac{1}{\sqrt{3}} \end{pmatrix}, \quad M_2^{[s]} = \begin{pmatrix} 0 & 0 \\ -\sqrt{\frac{2}{3}} & 0 \end{pmatrix}. \tag{2.12}$$

2.3 Gauge and normalization

The MPS representation of a quantum state is not unique. If $\{M_i^{[s]}\}_{i=1,\dots,d}^{s=1,\dots,N}$ is an MPS representation for vector $|\psi\rangle$, as in (2.3), then we can generate an infinite number of other representations $\{\widetilde{M}_i^{[s]}\}_{i=1,\dots,d}^{s=1,\dots,N}$, in this way:

$$\widetilde{M}_i^{[s]} = (X^{[s-1]})^{-1} M_i^{[s]} X^{[s]}, \quad \forall s \tag{2.13}$$

with $\{X^{[s]}\}^{s=1,\ldots,N}$ being any set of invertible matrices: $\exists(X^{[s]})^{-1}$, $\forall s$; the X matrices are squared with dimensions $D_s \times D_s$. We basically inserted identities between every two neighboring matrices and then we wrote them as $X^{-1}X$. We say that there is a *gauge freedom* in the choice of the MPS matrices. This freedom is exploited for very useful simplifications in the calculations. Once a particular set of matrices has been chosen we call that set: a *gauge*.

The normalization of an MPS presents some difficulties: suppose we know $|\psi\rangle$ has norm 2 and we want to rescale it in order to obtain unitary norm. Usually this reduces to the multiplication of every coefficient of $|\psi\rangle$ by $1/2$, but in an MPS every coefficient is formed by many matrices. The factor $1/2$ could be distributed equally over all matrices ($1/\sqrt[N]{2}$ each one) or assigned to only one matrix, for example. This generates different gauges. We decide to use a slightly different meaning for normalization: every choice of gauge defines a different *normalization* and the matrices in a given gauge are said *normalized* (with respect to that gauge). Moreover, we also call normalization any constraint or condition that MPS matrices must respect. It is significant that the normalization is not related to the single matrix but to the whole site. Indeed, you cannot choose a different X for each matrix of the site, you can only get one per site.

Gauges are useful to choose a single representative MPS among all the possible alternatives for a single state. The most important gauges we are going to talk about here are the *isometric gauges* that can be generated only for OBC lattices. They are used to define the canonical forms of MPS representations presented in the next three subsections.

2.3.1 Left-canonical MPS

Take an OBC MPS with random gauge, as in (2.3), and start from the leftmost matrices. Join vertically the matrices $\{M_i^{[1]}\}_i$ in a single matrix $M_V^{[1]}$ of dimensions $d \times D_1$. Then, applying an SVD (see appendix A), we obtain the usual three matrices:

$$M_V^{[1]} = U^{[1]}\Sigma^{[1]}V^{[1]\dagger}. \tag{2.14}$$

Matrix $U^{[1]}$ has again d rows and we can split it back into d row vectors that we call $A_i^{[1]}$. The remaining $\Sigma^{[1]}V^{[1]\dagger}$ may be multiplied to the right to *each one* of the matrices of site 2. In this way:

$$M_{i_1}^{[1]}M_{i_2}^{[2]}\cdots M_{i_N}^{[N]} = M_V^{[1]}M_{i_2}^{[2]}\cdots M_{i_N}^{[N]} \tag{2.15}$$

$$= U^{[1]}\left(\Sigma^{[1]}V^{[1]\dagger}M_{i_2}^{[2]}\right)\cdots M_{i_N}^{[N]} \tag{2.16}$$

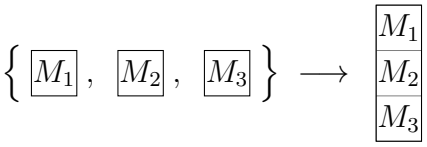


Figure 2.2: The process of grouping a set of matrices one below the other in a single taller matrix.

$$= A^{[1]}_{i_1} \widetilde{M}^{[2]}_{i_2} \cdots M^{[N]}_{i_N}. \tag{2.17}$$

The process is repeated on site 2: join together matrices $\{\widetilde{M}^{[2]}\}_i$ in a single matrix $M^{[2]}_V$, adding the first row of $\widetilde{M}^{[2]}_2$ under the last row of $\widetilde{M}^{[2]}_1$ and so on (see figure 2.2). Again we apply SVD and we obtain:

$$A^{[1]}_{i_1} \widetilde{M}^{[2]}_{i_2} M^{[3]}_{i_3} \cdots M^{[N]}_{i_N} = A^{[1]}_{i_1} M^{[2]}_V M^{[3]}_{i_3} \cdots M^{[N]}_{i_N} \tag{2.18}$$

$$= A^{[1]}_{i_1} U^{[2]} \left(\Sigma^{[1]} V^{[1]\dagger} M^{[3]}_{i_3} \right) \cdots M^{[N]}_{i_N} \tag{2.19}$$

$$= A^{[1]}_{i_1} A^{[2]}_{i_2} \widetilde{M}^{[3]}_{i_3} \cdots M^{[N]}_{i_N}. \tag{2.20}$$

In the last line, we divided $U^{[2]}$ into d matrices $\{A^{[2]}_i\}_i$ with shape $D_1 \times D_2$. The index i_2 momentarily disappears as it gets absorbed in the index of the rows of $U^{[2]}$.

After N steps, the matrices in all sites have been substituted:

$$|\psi\rangle = \sum_{i_1,\ldots,i_N=1}^{d} A^{[1]}_{i_1} \cdots A^{[N]}_{i_N} |i_1 \cdots i_N\rangle . \tag{2.21}$$

We have dropped the last residual pair of matrices, $\Sigma^{[N]} V^{[N]\dagger}$, that at this point are just a pair of coefficients. These A matrices are obtained from the splitting of the columns of isometric matrices and, thus, they have to respect the following crucial condition:

$$\sum_{i=1}^{d} A^{[s]\dagger}_i A^{[s]}_i = \mathbb{1}, \quad \forall s, \tag{2.22}$$

called *left-normalization*. We went through all that trouble just to obtain this identity, because we will see that it is very useful. When all the MPS matrices satisfy (2.22), we say that the MPS is in *left-canonical* form.

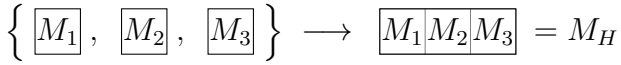


Figure 2.3: The process of grouping a set of matrices one on the right of the other in a single wider matrix.

2.3.2 Right-canonical MPS

An obvious dual gauge is found replacing all the matrices with the $V^\dagger$-part of SVD instead of the U-part. This time the operations have to start from the rightmost matrices and the grouping is column-wise instead of row-wise (see figure 2.3). Join $\{M_i^{[N]}\}_i$ into $M_H^{[N]}$ of dimensions $D_{N-1} \times d$ with every column-vector on the right of the previous one. SVD extract the unitary portion and the rest is multiplied to the left:

$$M_{i_1}^{[1]} \cdots M_{i_{N-1}}^{[N-1]} M_{i_N}^{[N]} = M_{i_1}^{[1]} \cdots M_{i_{N-1}}^{[N-1]} M_H^{[N]} \tag{2.23}$$

$$= M_{i_1}^{[1]} \cdots \left(M_{i_{N-1}}^{[N-1]} U^{[N]} \Sigma^{[N]} \right) V^{[N]\dagger} \tag{2.24}$$

$$= M_{i_1}^{[1]} \cdots \widetilde{M}_{i_{N-1}}^{[N-1]} B_{i_N}^{[N]}. \tag{2.25}$$

Matrix $V^{[N]\dagger}$ has been split back into separate columns B_i.

Going through all the matrices backwards, in the end we obtain a *right-canonical* MPS:

$$|\psi\rangle = \sum_{i_1,\ldots,i_N=1}^{d} B_{i_1}^{[1]} \cdots B_{i_N}^{[N]} |i_1 \cdots i_N\rangle , \tag{2.26}$$

where all the matrices satisfy the *right-normalization* condition:

$$\sum_{i=1}^{d} B_i^{[s]} B_i^{[s]\dagger} = \mathbb{1}, \quad \forall s. \tag{2.27}$$

2.3.3 Mixed-canonical MPS

Left- and right-normalization do just fine when they are mixed, in fact every site could have a random normalization. In practice, it is useful in a lot of situations when all the matrices on the left (until site s) are left-normalized and all the matrices on the right (from site $t > s$) are right-normalized. The string of matrices for every coefficient looks something like this:

$$A^{[1]} A^{[2]} \cdots A^{[s]} M^{[s+1]} \cdots M^{[t-1]} B^{[t]} \cdots B^{[N-1]} B^{[N]}; \tag{2.28}$$

we omitted the lower indices i_s to stress that normalization is a property of the site and not of the single matrix. Letter M specifies an unknown normalization for that site.

Such a configuration is easy to obtain applying the left-normalization process from site 1 to site s and the right-normalization process from site N to site t (backwards). The exceeding matrices $\Sigma^{[s]}V^{[s]\dagger}$ and $U^{[t]}\Sigma^{[t]}$ that remain in the end are absorbed on the left and on the right, respectively, by $M^{[s+1]}$ and $M^{[t-1]}$.

The limit case is when $t = s+1$: this arrangement has a special meaning and a very important role. Say we already left-normalized all sites from 1 to s and also right-normalized all sites from N to $s+2$. The delicate step is the last one, the right-normalization of site $s+1$:

$$A^{[1]} \cdots A^{[s]} M^{[s+1]} B^{[s+2]} \cdots B^{[N]}. \tag{2.29}$$

We obtain $M^{[s+1]} = U^{[s+1]}\Sigma^{[s+1]}B^{[s+1]}$, now $B^{[s+1]}$ replaces $M^{[s+1]}$, but if we absorb $U^{[s+1]}\Sigma^{[s+1]}$ into $A^{[s]}$ we spoil the left-normalization of the latter. Instead we absorb only $U^{[s+1]}$ while the diagonal matrix remains explicit. We define $\widetilde{A}^{[s]} := A^{[s]}U^{[s+1]}$ and, by the isometric property of U, we obtain:

$$\sum_{i=1}^{d} \widetilde{A}_i^{[s]\dagger}\widetilde{A}_i^{[s]} = U^{[s+1]\dagger}\left(\sum_{i=1}^{d} A_i^{[s]\dagger}A_i^{[s]}\right)U^{[s+1]} \tag{2.30}$$

$$= U^{[s+1]^\dagger}\mathbb{1}U^{[s+1]} = \mathbb{1}, \tag{2.31}$$

i.e., $\widetilde{A}^{[s]}$ is still left-normalized. When sites $1, \ldots, s$ are left-normalized, sites $s+1, \ldots, N$ are right-normalized and, consequently, a diagonal matrix remains explicit on bond s, we say that the MPS is in *mixed-canonical* form with *central bond* s:

$$|\psi\rangle = \sum_{i_1,\ldots,i_N=1}^{d} A_{i_1}^{[1]} \cdots A_{i_s}^{[s]}\Sigma^{[s]}B_{i_{s+1}}^{[s+1]} \cdots B_{i_N}^{[N]} |i_1 \cdots i_N\rangle . \tag{2.32}$$

The mixed-canonical form has a special physical connection: from it we are able to read directly the Schmidt decomposition of the state (see section 1.3) and, then, to calculate many important properties of the system, like entanglement. Consider a mixed-canonical MPS with central bond s, if we multiply together only the left-normalized matrices we do not obtain a single number (unless $s = N$), but a horizontal vector. For every element of the vector we define a new state:

$$|e_k\rangle_A = \sum_{i_1,\ldots,i_s}^{d} \left(A_{i_1}^{[1]} \cdots A_{i_s}^{[s]}\right)_{1,k} |i_1 \cdots i_s\rangle , \tag{2.33}$$

making a total of D_s states. What is important about them is that they are orthonormal thanks to the normalization property:

$$
\begin{aligned}
{}_A\langle e_{k'}|e_k\rangle_A &= \sum_{i_1,\dots,i_s}^{d} \left(A^{[1]}_{i_1}\cdots A^{[s]}_{i_s}\right)^*_{1,k'} \left(A^{[1]}_{i_1}\cdots A^{[s]}_{i_s}\right)_{1,k} && (2.34)\\
&= \sum_{i_1,\dots,i_s}^{d} \left(A^{[1]}_{i_1}\cdots A^{[s]}_{i_s}\right)^\dagger_{k',1} \left(A^{[1]}_{i_1}\cdots A^{[s]}_{i_s}\right)_{1,k} && (2.35)\\
&= \sum_{i_1,\dots,i_s}^{d} \left(A^{[s]\dagger}_{i_s}\cdots A^{[1]\dagger}_{i_1} A^{[1]}_{i_1}\cdots A^{[s]}_{i_s}\right)_{k',k} = \delta_{k',k}. && (2.36)
\end{aligned}
$$

The same is valid for those states constructed from the right-normalized matrices:

$$
|f_k\rangle_B = \sum_{i_{s+1},\dots,i_N}^{d} \left(B^{[s+1]}_{i_{s+1}}\cdots B^{[N]}_{i_N}\right)_{k,1} |i_{s+1}\cdots i_N\rangle\,, \qquad (2.37)
$$

note that here we took the coefficients from the elements of a column vector. In turns, this implies that, when we take the overlap of two states $|f_k\rangle_B$, the two corresponding coefficients must be swapped (with respect to equation (2.34)). The adjoint matrices will then be on the right side thus making the right-normalization the proper choice for orthonormality:

$$
\begin{aligned}
{}_B\langle f_{k'}|f_k\rangle_B &= \sum_{i_{s+1},\dots,i_N}^{d} \left(B^{[s+1]}_{i_{s+1}}\cdots B^{[N]}_{i_N}\right)^*_{k',1} \left(B^{[s+1]}_{i_{s+1}}\cdots B^{[N]}_{i_N}\right)_{k,1} && (2.38)\\
&= \sum_{i_{s+1},\dots,i_N}^{d} \left(B^{[n+1]}_{i_{s+1}}\cdots B^{[N]}_{i_N}\right)_{k,1} \left(B^{[s+1]}_{i_{s+1}}\cdots B^{[N]}_{i_N}\right)^\dagger_{1,k'} && (2.39)\\
&= \sum_{i_{s+1},\dots,i_N}^{d} \left(B^{[n+1]}_{i_{s+1}}\cdots B^{[N]}_{i_N} B^{[N]\dagger}_{i_N}\cdots B^{[s+1]\dagger}_{i_{s+1}}\right)_{k,k'} = \delta_{k,k'}. && (2.40)
\end{aligned}
$$

To emphasize this property, the central bond of a mixed-canonical MPS is often called *orthonormality center*.

For the final step, we regroup the matrices of the mixed-canonical state (2.32) in order to explicitly show the states $|e_k\rangle_A$ and $|f_k\rangle_B$; remember that Σ is a diagonal matrix, with diagonal elements σ_k:

$|\psi\rangle =$

$$\sum_{k,k'=1}^{D_s} \sum_{i_1,\ldots,i_N=1}^{d} \left(A_{i_1}^{[1]} \cdots A_{i_s}^{[s]}\right)_{1,k} \left(\Sigma^{[s]}\right)_{k,k'} \left(B_{i_{s+1}}^{[s]} \cdots B_{i_N}^{[N]}\right)_{k',1} |i_1 \cdots i_N\rangle$$
$$= \sum_{k=1}^{D_s} \sigma_k |e_k\rangle_A |f_k\rangle_B, \quad (2.41)$$

clearly, this is the Schmidt decomposition of $|\psi\rangle$ according to the bipartition $1, \ldots, s : s{+}1, \ldots, N$. This is possible only when the state has orthonormality center at s.

2.4 Valence bond derivation

In the light of what we have seen in section 1.5.5, we know that ground states of gapped, local hamiltonians in 1D must have entanglement bounded by a constant. In this section, we want to explain how to construct a state that automatically satisfy an area law and how the set of these states is parametrized by matrix product states. We want to emphasize that the procedure is quite remarkable and it is at the basis of many developments in the field [53, 54].

With the same notation of section 2.1, consider a finite 1D lattice with N sites and periodic boundary conditions (PBC). In every site there is a system that can be described by a d-dimensional Hilbert space. Now suppose that we substitute every (physical) system with a *virtual* pair of particles each described by a D-dimensional Hilbert space (with D finite and fixed for every site).

We choose the state in which these $2N$ D-dimensional systems are as follows: numbering the sites from 1 to N and labeling each pair with A and B, we link together every particle B with particle A in the adjacent site on the right with a maximally entangled state as:

$$|\chi_D\rangle_{sB,(s+1)A} = \frac{1}{\sqrt{D}} \sum_{j=1}^{D} |j\rangle_{sB} |j\rangle_{(s+1)A}, \quad (2.42)$$

see figure 2.4. Particle B in the last site will form a maximally entangled pair with particle A in the first site, as imposed by PBC.

This kind of state is called *valence bond solid*; the total state is:

$$|\chi_D\rangle^{\otimes N} = \left(\bigotimes_{s=1}^{N-1} |\chi_D\rangle_{sB,(s+1)A}\right) |\chi_D\rangle_{NB,1A}. \quad (2.43)$$

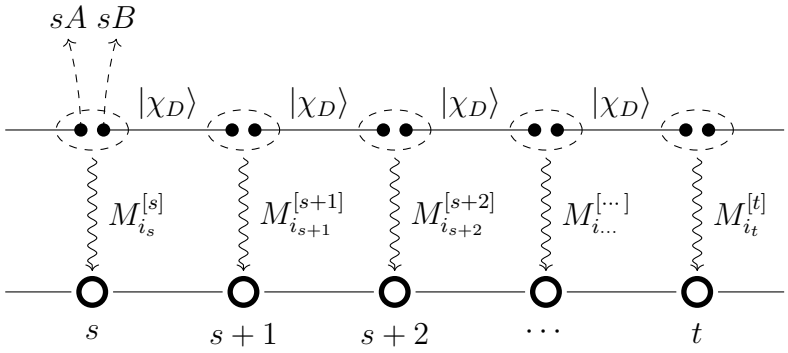


Figure 2.4: Valence bond derivation of matrix product states: in the upper row we have the pairs AB of D-dimensional virtual particles; in the lower row there are the d-dimensional physical particles. The virtual pairs are linked by a maximally entangled state $|\chi_D\rangle$ between adjacent sites. The state of the virtual pair of site s is mapped to a state of the physical system in site s through a linear function that depends on the choice of the matrix $M_{i_s}^{[s]}$

It is not too difficult to guess that the entanglement content of this state is limited and what that limit is. Without loss of generality, take the bipartition $1\ldots n : n+1\ldots N$, then we must determine the reduced density matrix:

$$\rho_{1\ldots n} = \mathrm{Tr}_{n+1\ldots N}\Big[\, |\chi_D\rangle^{\otimes N} \langle\chi_D|^{\otimes N} \,\Big]. \tag{2.44}$$

Remembering the definition of partial trace (equation (1.10)), we obtain:

$$\rho_{1\ldots n} = \rho_{1A}^{\max} \otimes |\chi_D\rangle_{1B,2A}\langle\chi_D| \otimes \cdots \otimes |\chi_D\rangle_{(n-1)B,nA}\langle\chi_D| \otimes \rho_{nB}^{\max}, \tag{2.45}$$

where

$$\rho_A^{\max} = \mathrm{Tr}_B[\,|\chi_D\rangle\langle\chi_D|\,] = \frac{1}{D}\sum_{j=1}^{D} |j\rangle_B\langle j| \tag{2.46}$$

is a state that maximizes von Neumann entropy (see section 1.2).

Using the fact that $S(\rho_1 \otimes \rho_2) = S(\rho_1) + S(\rho_2)$, we have that the entropy of entanglement is:

$$E_E(|\chi_D\rangle^{\otimes N} \langle\chi_D|^{\otimes N}) = S(\rho_{1\ldots n}) = S(\rho_{1A}^{\max}) + S(\rho_{nB}^{\max}) = 2\log_2 D, \tag{2.47}$$

because von Neumann entropy vanishes for pure states. So, for any choice of bipartition, the entropy of entanglement for this state is bounded.

To return from the virtual particles space to the physical space, we apply to each site the linear transformation:

$$P^{[s]} = \sum_{i_s=1}^{d}\sum_{\alpha,\beta=1}^{D} (M_{i_s}^{[s]})_{\alpha\beta}\, |i_s\rangle_{\,sA,sB}\langle\alpha,\beta|; \tag{2.48}$$

in this way we can map $|\chi_D\rangle^{\otimes N}$ to a suitable state for the physical system:

$$|\chi_D\rangle^{\otimes N} \longrightarrow |\psi\rangle = P^{\otimes N} |\chi_D\rangle^{\otimes N}. \tag{2.49}$$

We apply only one map P to see what the result is:

$$P^{[2]}|\chi_D\rangle_{1B,2A}|\chi_D\rangle_{2B,3A} = \sum_{i_2=1}^{d}\sum_{j,k=1}^{D}(M^{[2]}_{i_2})_{jk}\,|i_2\rangle\,(|j\rangle_{1B}|k\rangle_{3A}). \tag{2.50}$$

Then we add another P and another entangled state to the right:

$$\begin{aligned} P^{[2]}P^{[3]}|\chi_D\rangle_{1B,2A}|\chi_D\rangle_{2B,3A}|\chi_D\rangle_{3B,4A} = \\ = \sum_{i_2,i_3=1}^{d}\sum_{j,k,l=1}^{D}(M^{[2]}_{i_2})_{j,k}(M^{[3]}_{i_3})_{k,l}\,|i_2 i_3\rangle\,(|j\rangle_{1B}|l\rangle_{4A}) \\ = \sum_{i_2,i_3=1}^{d}\sum_{j,l=1}^{D}(M^{[2]}_{i_2}M^{[3]}_{i_3})_{jl}\,|i_2 i_3\rangle\,(|j\rangle_{1B}|l\rangle_{4A}). \end{aligned} \tag{2.51}$$

We absorbed the sum over index k in the matrix multiplication. It is easy to see how this will continue; applying the same passage over and over until $P^{[N]}$, we obtain:

$$\begin{aligned} P^{[2]}\cdots P^{[N]}\,|\chi_D\rangle^{\otimes N} = \\ = \sum_{i_2,\ldots,i_N=1}^{d}\sum_{j,l=1}^{D}(M^{[2]}_{i_2}\cdots M^{[N]}_{i_N})_{jl}\,|i_2\cdots i_N\rangle\,(|j\rangle_{1B}|l\rangle_{1A}); \end{aligned} \tag{2.52}$$

and finally we act with $P^{[1]}$ that contracts indices j and l too, obtaining a trace over all matrices:

$$|\psi\rangle = P^{\otimes N}\,|\chi_D\rangle^{\otimes N} = \sum_{i_1,\ldots,i_N}^{d}\mathrm{Tr}\Big[\,M^{[1]}_{i_1}\cdots M^{[N]}_{i_N}\,\Big]\,|i_1\cdots i_N\rangle\,. \tag{2.53}$$

This is exactly the MPS form for PBC that we introduced in section 2.1.

Note that every transformation P is local to the site it acts on and therefore $P^{\otimes N}$ is local for every possible bipartition of the chain. This is why we are sure that the entanglement between any bipartition cannot increase:

$$E_E(|\psi\rangle\langle\psi|) = S(\rho_{1\ldots n}) \leq 2\log_2 D, \quad \forall n. \tag{2.54}$$

In the case of open boundary conditions (OBC), we loose a maximally entangled pair, dropping the virtual particles $1A$ and NB in the first and

last site. The matrices that define the linear transformations for those sites must be redefined:

$$P^{[1]} = \sum_{i_1=1}^{d} \sum_{\beta=1}^{D} (M_{i_1}^{[1]})_\beta \, |i_1\rangle_{\,1B} \langle \beta| \tag{2.55}$$

$$P^{[N]} = \sum_{i_N=1}^{d} \sum_{\alpha=1}^{D} (M_{i_N}^{[N]})_\alpha \, |i_N\rangle_{\,NA} \langle \alpha|; \tag{2.56}$$

we have that $M_{i_1}^{[1]}$ are row vectors while $M_{i_N}^{[N]}$ are column vectors. So there is no need for the trace in the physical state because the result of the product of all the matrices is always a number:

$$|\psi\rangle = P^{\otimes N} |\chi_D\rangle^{\otimes N-1} = \sum_{i_1,\ldots,i_N}^{d} M_{i_1}^{[1]} \cdots M_{i_N}^{[N]} \, |i_1 \cdots i_N\rangle \, . \tag{2.57}$$

As a last remark, the dimension D does not need to remain fixed for all the sites, we opted for a constant value for sake of clarity. On the contrary, the procedure described above is still valid with a changing D_s for every site s, so to recover the full notation for MPS that we introduced in section 2.1. Moreover, as long as the D_s does not increase with the system size N, the area law of equation (2.54) is still valid.

2.5 SVD derivation

We know that every MPS is a well defined vector; here we are going to show that every vector has an MPS form. We will do this, once again, with the help of SVD (see appendix A). The procedure is almost always numerically impractical, but it is formally correct.

Take a generic vector that lives in a d^N-dimensional Hilber space:

$$|\psi\rangle = \sum_{i_1,\ldots,i_N=1}^{d} c_{i_1 \cdots i_N} \, |i_1 \cdots i_N\rangle \, , \tag{2.58}$$

we can always reshape the coefficients $c_{i_1 \cdots i_N}$ into a matrix $\Psi^{[1]}$ of dimensions $d \times d^{N-1}$ whose elements are defined as follows:

$$(\Psi^{[1]})_{i_1,(i_2,\ldots,i_N)} \coloneqq c_{i_1 i_2 \cdots i_N} . \tag{2.59}$$

The regrouping of indices $(i_2, \ldots, i_N)$ is a common practice in the manipulation of matrices[1].

We apply the reduced SVD:

$$\Psi^{[1]} = U^{[1]}\Sigma^{[1]}V^{[1]\dagger}; \tag{2.61}$$

that, with explicit indices, means:

$$(\Psi^{[1]})_{i_1,(i_2,\ldots,i_N)} = \sum_{\alpha_1=1}^{d} u^{[1]}_{i_1\alpha_1}\sigma^{[1]}_{\alpha_1}v^{[1]\dagger}_{\alpha_1,(i_2,\ldots,i_N)} = \sum_{\alpha_1=1}^{d}(A^{[1]}_{i_1})_{\alpha_1}(\Psi^{[2]})_{(\alpha_1,i_2),(i_3,\ldots,i_N)}. \tag{2.62}$$

We split matrix U into d row vectors $A^{[1]}_i$ of dimensions $1 \times d$. These will be the matrices for the first site. We also reshaped what remains on the right hand side into a new matrix $\Psi^{[2]}$, this time of dimensions $d^2 \times d^{N-2}$:

$$(\Psi^{[2]})_{(\alpha_1,i_2),(i_3,\ldots,i_N)} \coloneqq \sigma^{[1]}_{\alpha_1}v^{[1]\dagger}_{\alpha_1,(i_2,\ldots,i_N)}. \tag{2.63}$$

note that we moved index i_2 on the left among the row indices.

Again we apply a reduced SVD to $\Psi^{[2]}$; our goal is to separate index i_2 to another matrix:

$$(\Psi^{[2]})_{(\alpha_1,i_2),(i_3,\ldots,i_N)} = \sum_{\alpha_2=1}^{d^2} u^{[2]}_{(\alpha_1,i_2),\alpha_2}\sigma^{[2]}_{\alpha_2}v^{[2]\dagger}_{\alpha_2,(i_3,\ldots,i_N)}. \tag{2.64}$$

We reshape $U^{[2]}$ into d matrices of dimensions $d \times d^2$:

$$(A^{[2]}_i)_{\alpha_1\alpha_2} \coloneqq u^{[2]}_{(\alpha_1,i),\alpha_2} \tag{2.65}$$

and $\Sigma^{[2]}V^{[2]\dagger}$ into $\Psi^{[3]}$ of dimensions $d^2 \times d^{N-3}$:

$$(\Psi^{[3]})_{(\alpha_2 i_3),(i_4,\ldots,i_N)} \coloneqq \sigma^{[2]}_{\alpha_2}v^{[2]\dagger}_{\alpha_2,(i_3,\ldots,i_N)}. \tag{2.66}$$

Substituting everything in (2.62) we obtain:

$$c_{i_1\cdots i_N} = \sum_{\alpha_1=1}^{d}\sum_{\alpha_2=1}^{d^2}(A^{[2]}_{i_1})_{\alpha_1}(A^{[2]}_{i_2})_{\alpha_1\alpha_2}(\Psi^{[3]})_{(\alpha_2 i_3),(i_4,\ldots,i_N)}. \tag{2.67}$$

[1]For example, the regrouping of three indices (j_1, j_2, j_3) defines a new index k with this simple conversion formula, supposing indices j assume values in $\{1, \ldots, d\}$:

$$(j_1, j_2, j_3) \longrightarrow k = (j_1 - 1)d^2 + (j_2 - 1)d + j_3. \tag{2.60}$$

then index k will correctly assume values between 1 and d^3.

1	2	$\cdots$	$\frac{N}{2}$	$\frac{N}{2}+1$	$\cdots$	$N-1$	N
$1 \times d$	$d \times d^2$	$\cdots$	$d^{\frac{N}{2}-1} \times d^{\frac{N}{2}}$	$d^{\frac{N}{2}} \times d^{\frac{N}{2}-1}$	$\cdots$	$d^2 \times d$	$d \times 1$

Table 2.1: Dimension for every site of the matrices generated by the SVD derivation of MPS; the upper row indicates the number of the site.

Iterating these steps, it is easy to see how the MPS form emerges. When we reach the final index we obtain:

$$c_{i_1 \cdots i_N} = \sum_{\alpha_1=1}^{d} \sum_{\alpha_2=1}^{d^2} \cdots \sum_{\alpha_{N-1}=1}^{d} (A^{[1]}_{i_1})_{\alpha_1} (A^{[2]}_{i_2})_{\alpha_1 \alpha_2} \cdots (A^{[N]}_{i_N})_{\alpha_{N-1}} \tag{2.68}$$

$$= A^{[1]}_{i_1} A^{[2]}_{i_2} \cdots A^{[N]}_{i_N}. \tag{2.69}$$

The MPS obtained is already in left-canonical form. An analogous procedure could start from the rightmost matrices, taking the sliced $V^\dagger$ parts of the SVD as the matrices of the MPS, defining a right-canonical form.

What we presented is both a new method to introduce MPSs and a proof that every tensor of every dimensionality has an MPS form. Indeed, it is sufficient to choose a numbering of the sites of the lattice to decompose the tensor defined by the coefficients of the state into smaller blocks, formed by groups of matrices.

Another important aspect of this derivation is that, starting from a very generic setting, we are able to learn something about the distribution of quantum correlations in a chain. In particular, as we can see from table 2.1, the size of the matrices increases exponentially until the middle of the chain. From section 2.4, we know that the entropy of entanglement of a bipartition of the system cannot be more than the sum of the logarithms of the bond link dimensions that separate the two parts (equation (2.54)). This, in turns, means that entanglement grows linearly until the two parts of the system are equal: both a half of the chain. In the thermodynamic limit, sending N to infinity, we see that the entanglement is proportional to the size of the smaller of the parts, and we recover the fact that a generic many-body state may respect a volume law (section 1.5.4):

$$S(\rho_{1\ldots n}) = \log_2 D_n = \log_2 d^n = n \log_2 d. \tag{2.70}$$

To satisfy an area law, it means that the state has bond link dimensions D_s bounded by a constant and this constant do not depends on N. Usually, we impose a cutoff D to the matrices so that their maximum dimension is $D \times D$. We pass then from $\mathcal{O}(d^N)$ parameters, exponential in the number of

sites, to $\mathcal{O}(NdD^2)$ parameters, linear in the number of sites, thus, the state can be saved on a memory efficiently.

2.6 Graphical notation

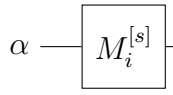
MPS calculations often involve a great number of tensors and indices, too much to quickly figure out what is going on. To overcome these difficulties, a graphical notation has been introduced. The importance is twofold:

- it simplifies the visualisation and validation of complex operations with too many symbols and indices;
- it helps the writing of abstract algorithms and concrete implementations on a machine and also the reconstruction of the original formula behind the graph.

The idea is loosely based on Penrose's tensor notation where a tensor is represented by a geometrical shape with a number a segments (or *legs*) pointing out equal to the number of indices the tensor has. A contraction between two indices is converted into a line that connects the corresponding two legs.

2.6.1 Graph elements

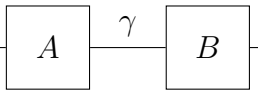
Matrices. Since a matrix is just a tensor with two indices we will represent it graphically as a box with two legs pointing left and right. By convention, we assume that the left leg specifies the row number, while the right leg the column number. Take for example the i-th matrix of site s of an MPS, its elements are $(M_i^{[s]})_{\alpha\beta}$ and it is represented as:



(2.71)

A multiplication between matrices A and B translates to:

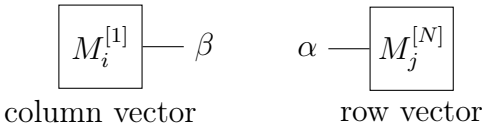
$$(C)_{\alpha\beta} = \sum_{\gamma} (A)_{\alpha\gamma}(B)_{\gamma\beta} : \qquad$$



(2.72)

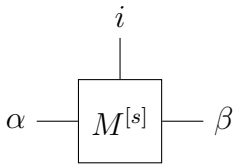
Vectors. Vectors have only one index, so only one leg; the shape is still a box. If it is a column vector then the leg points right, while if it is a row

vector the leg points left. As an example, we take elements from the first and the last site of an OBC MPS:



(2.73)

3-tensors. Every site of an MPS has three indices so we represent it as a 3-tensor, namely as a box with three legs:



(2.74)

Scalars. A scalar is a special case of a matrix: one with no indices, so we illustrate it as a box with no legs. We will make use of no scalars just as single boxes, but note that every graph with no legs pointing out (i.e. with all the legs connected and all the indices contracted) *is* a scalar. For example, this is the overlap between two simple vectors:

$$\sum_\alpha (v_1)^*_\alpha (v_2)_\alpha : \qquad \boxed{v_1^*} \overset{\alpha}{\text{---}} \boxed{v_2} \ . \tag{2.75}$$

In the same way, every graph with only one unpaired leg is a vector, every graph with only two free legs is a matrix and so on. This is how the graphical notation is very helpful: we immediately recognize the kind of object we are dealing with.

Straight lines and diagonal matrices. As already mentioned, every straight line that links two boxes is an index contraction. On the other hand, we can look at it as a representation of the Kronecher delta or of the identity matrix:

$$\lambda \overset{\delta_{\lambda\lambda'}}{\text{---}} \lambda' \ . \tag{2.76}$$

Moreover, the identity matrix is a special case of diagonal matrix; in fact, every diagonal matrix could be used as a contraction because it pairs the indices on the left and on the right. Unlike the identity matrix, a generic diagonal matrix weights the indices with different values. Here we give a

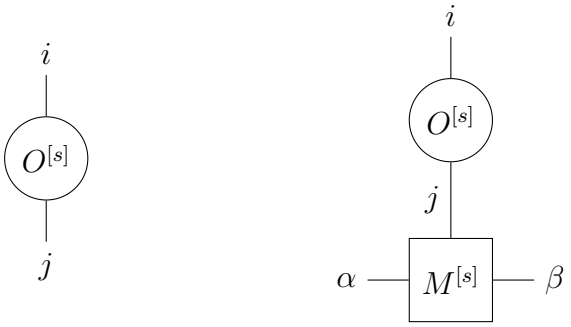
special role to diagonal matrices and we represent them differently from other matrices: with a diamond (or rotated-square) shape:

diagonal matrix (2.77)

Single-site operators. Operators acting on single sites appear very often in the problems that interest us, mainly because hamiltonians and other observables are composed of tensor products of them. They are the most simple kind of operators acting on MPS that we can think of, for example:

$$O^{[s]} = \sum_{i,j=1}^{d} f_j^i \, |i\rangle\langle j| \tag{2.78}$$

is an operator acting on site s. We illustrate them with a round shape and, since they have two *physical* indices, we add two legs in the vertical direction: one up and one down. Below we show, on the left, a single-site operator by itself and, on the right, the action of this operator on a 3-tensor, the result is obviously another 3-tensor:



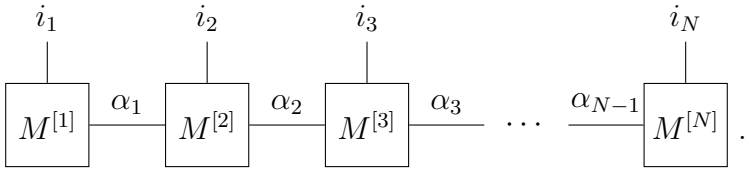
(2.79)

2.6.2 MPS graphs

Putting all these instructions together, we are now able to draw the graphical equivalent of a matrix product state. We begin with the OBC case, where first and last sites have row and column vectors. In the middle there are matrices all contracted in a multiplication and every site has an incoming physical index. Since no normalization is assumed, we represent this MPS as many boxes on a single line all connected in the horizontal direction and

with many legs pointing upwards:

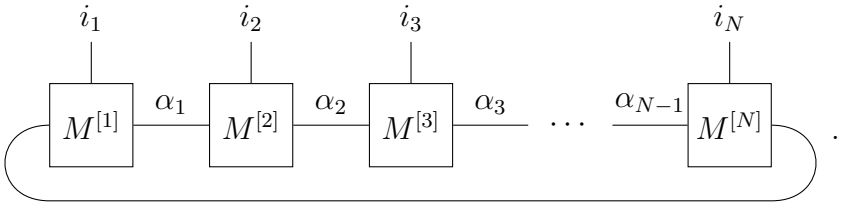
$$|\psi\rangle = \sum_{i_1,\ldots,i_N=1}^{d} M^{[1]}_{i_1} \cdots M^{[N]}_{i_N} |i_1 \cdots i_N\rangle :$$



(2.80)

The PBC case is not very different, first and last sites contain matrices and the spare legs are connected by the trace:

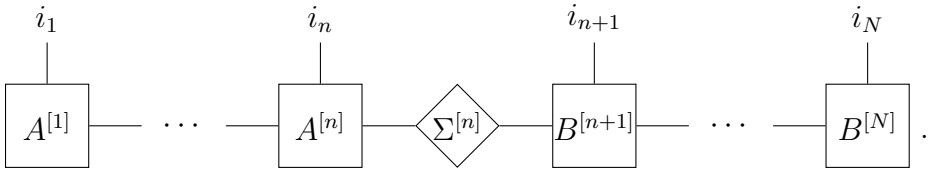
$$|\psi\rangle = \sum_{i_1,\ldots,i_N=1}^{d} \mathrm{Tr}\Big[M^{[1]}_{i_1} \cdots M^{[N]}_{i_N} \Big] |i_1 \cdots i_N\rangle :$$



(2.81)

There is no sign of distinction for site normalizations, apart from the letter used to name the matrices, so left-canonical and right-canonical MPSs are graphically equivalent. Anyway be aware that mixed-canonical MPSs usually have a diagonal matrix somewhere. This really helps with finding the orthonormality center at first glance:

$$|\psi\rangle = \sum_{i_1,\ldots,i_N=1}^{d} A^{[1]}_{i_1} \cdots A^{[n]}_{i_n} \Sigma^{[n]} B^{[n+1]}_{i_{n+1}} \cdots B^{[N]}_{i_N} |i_1 \cdots i_N\rangle :$$

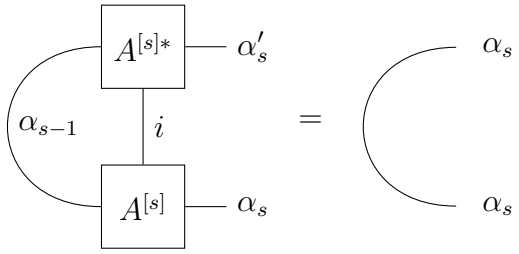


(2.82)

Finally, and this will be useful in the next section, we also represent graphically the meaning of left- and right-normalization conditions. Remembering

that an identity matrix is equivalent to a line:

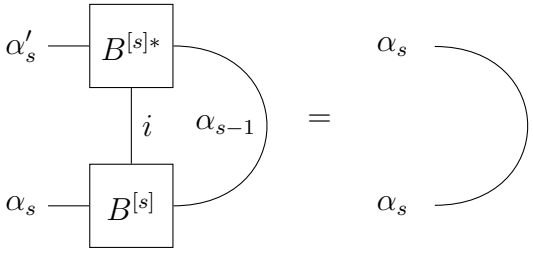
$$\sum_{i=1}^{d} A_i^{[s]\dagger} A_i^{[s]} = \mathbb{1} :$$



(2.83)

this for left-normalization, while for right-normalization:

$$\sum_{i=1}^{d} B_i^{[s]} B_i^{[s]\dagger} = \mathbb{1} :$$



(2.84)

2.7 Transfer operators for overlaps and expectation values

Consider two kets $|\psi\rangle$, $|\varphi\rangle$ and an operator O. Operations like overlaps (inner products: $\langle\varphi|\psi\rangle$), expectation values ($\langle\psi|O|\psi\rangle$) and matrix elements ($\langle\varphi|O|\psi\rangle$) are vectorial operations ubiquitous in quantum mechanics and also in quantum many-body theory. Moreover, they are very important operations in the most widely used algorithms dealing with MPSs as well as in the new algorithm we propose in section 3.2. In this section we would like to explain all these operations introducing the theory of transfer operators, that formalize them in the MPS context.

To correctly execute these operations it is necessary to pay attention not to follow the traditional way to compute them. Take for example the scalar product, normally the coefficients are multiplied two by two and then summed. The typical number of coefficients of a many body state vector grows exponentially with the size of the system. So the traditional way to compute the scalar product is inefficient: although we solved the problem of representing these giant vector states on a limited memory, now it seems impossible to make any useful operation with them. The solution is to choose a different order of contractions. In this way we exploit correctly the simplifications offered by the MPS representation. The important news is: the

order of contractions matters and, even if the contraction is already efficient, we should pay attention to the presence of faster schemes.

2.7.1 Overlaps

Take two OBC MPSs:

$$|\psi\rangle = \sum_{i_1,\dots,i_N=1}^{d} M_{i_1}^{[1]} \cdots M_{i_N}^{[N]} \, |i_1 \cdots i_N\rangle \,, \tag{2.85}$$

$$|\phi\rangle = \sum_{i_1,\dots,i_N=1}^{d} \widetilde{M}_{i_1}^{[1]} \cdots \widetilde{M}_{i_N}^{[N]} \, |i_1 \cdots i_N\rangle \,; \tag{2.86}$$

they have bond link dimensions of D and $\widetilde{D}$ respectively. We proceed analytically, taking the adjoint of $|\phi\rangle$ and applying it to $|\psi\rangle$ (see figure 2.5):

$$\langle\phi|\psi\rangle = \sum_{i_1,\dots,i_N=1}^{d} \left(\widetilde{M}_{i_1}^{[1]} \cdots \widetilde{M}_{i_N}^{[N]}\right)^* M_{i_1}^{[1]} \cdots M_{i_N}^{[N]}. \tag{2.87}$$

This formula, taken as it is, would require the calculation of $2d^N$ coefficients formed by strings of matrices all multiplied together. Instead, remembering that the complex conjugate of a number is equal to the adjoint of that number, consider the following regrouping:

$$\begin{aligned} \langle\phi|\psi\rangle &= \sum_{i_1,\dots,i_N=1}^{d} \widetilde{M}_{i_N}^{[N]\dagger} \cdots \widetilde{M}_{i_1}^{[1]\dagger} M_{i_1}^{[1]} \cdots M_{i_N}^{[N]} \\ &= \sum_{i_N=1}^{d} \widetilde{M}_{i_N}^{[N]\dagger} \left(\cdots \left(\sum_{i_2=1}^{d} \widetilde{M}_{i_2}^{[2]\dagger} \left(\sum_{i_1=1}^{d} \widetilde{M}_{i_1}^{[1]\dagger} M_{i_1}^{[1]} \right) M_{i_2}^{[2]} \right) \cdots \right) M_{i_N}^{[N]}; \end{aligned} \tag{2.88}$$

it means that first of all we multiply the matrices of the first site (it is an outer product) and then we sum over i_1. This produces the matrix $L^{[1]}$. In the next step, $\widetilde{M}_{i_2}^{[2]}$ multiplies $L^{[1]}$ and $M_{i_2}^{[2]}$, then all is summed over i_2. The process iterates with, at every step, two matrix products and a sum over matrix index, until the last site:

$$L^{[s]} = \sum_{i=1}^{d} \widetilde{M}_i^{[s]\dagger} L^{[s-1]} M_i^{[s]} \qquad L^{[0]} = 1. \tag{2.89}$$

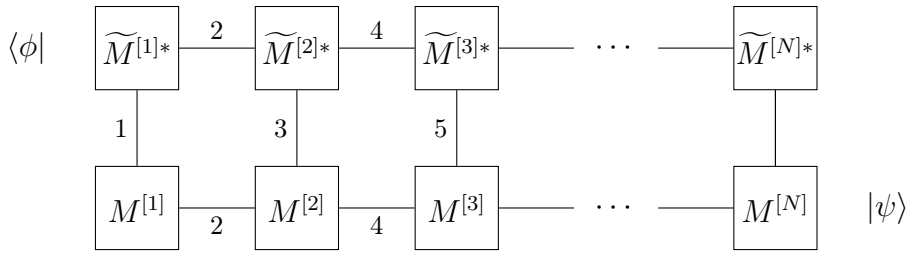


Figure 2.5: graph of the overlap between two OBC MPS, $|\psi\rangle$ and $|\phi\rangle$, with hinted order of contractions.

This is the optimal order of contractions hinted in figure 2.5 and it has a computational cost of

$$\mathcal{O}(NdD^2\widetilde{D}) + \mathcal{O}(NdD\widetilde{D}^2), \tag{2.90}$$

while the cost becomes $\mathcal{O}(NdD^3)$, if the two vectors have the same bond link dimension. Note that it is necessary that the two vectors have the same N and d but the bond link D_s could change for every site.

Overlaps can also be computed backwards, from the last site to the first. To do that, it is sufficient to exchange the two strings of matrix products in equation (2.87). The intermediate steps generate the matrices $R^{[s]}$:

$$R^{[s-1]} = \sum_{i=1}^{d} M_i^{[s]} R^{[s]} \widetilde{M}_i^{[s]\dagger} \qquad R^{[N]} = 1. \tag{2.91}$$

The square norm of $|\psi\rangle$, namely $\langle\psi|\psi\rangle$, is a special case of overlap. Here, the problems related to normalization, as expressed in section 2.3, return, because the result depends on the choice of gauge that we make for every single matrix. Anyway, if we choose all left-normalized matrices we see that equation (2.89) always corresponds to the normalization condition (2.22):

$$L^{[1]} = \sum_{i=1}^{d} A_i^{[1]\dagger} A_i^{[1]} = \mathbb{1}, \quad L^{[2]} = \sum_{i=1}^{d} A_i^{[2]\dagger} \mathbb{1} A_i^{[2]} = \mathbb{1}, \quad \dots \tag{2.92}$$

so we immediately obtain that $\langle\psi|\psi\rangle = 1$.

2.7.2 Transfer operators

The operations that transform $L^{[s-1]}$ to $L^{[s]}$ or $R^{[s]}$ to $R^{[s-1]}$ map matrices to other matrices. We formalize these actions defining an *ad-hoc* (super)operator $\mathbb{E}^{[s]}_{\langle\phi|\psi\rangle}$ for overlap $\langle\phi|\psi\rangle$ and site s:

$$\mathbb{E}^{[s]}_{\langle\phi|\psi\rangle} : \mathbb{C}^{\widetilde{D}_{s-1}\times D_{s-1}} \to \mathbb{C}^{\widetilde{D}_s \times D_s}, \tag{2.93}$$

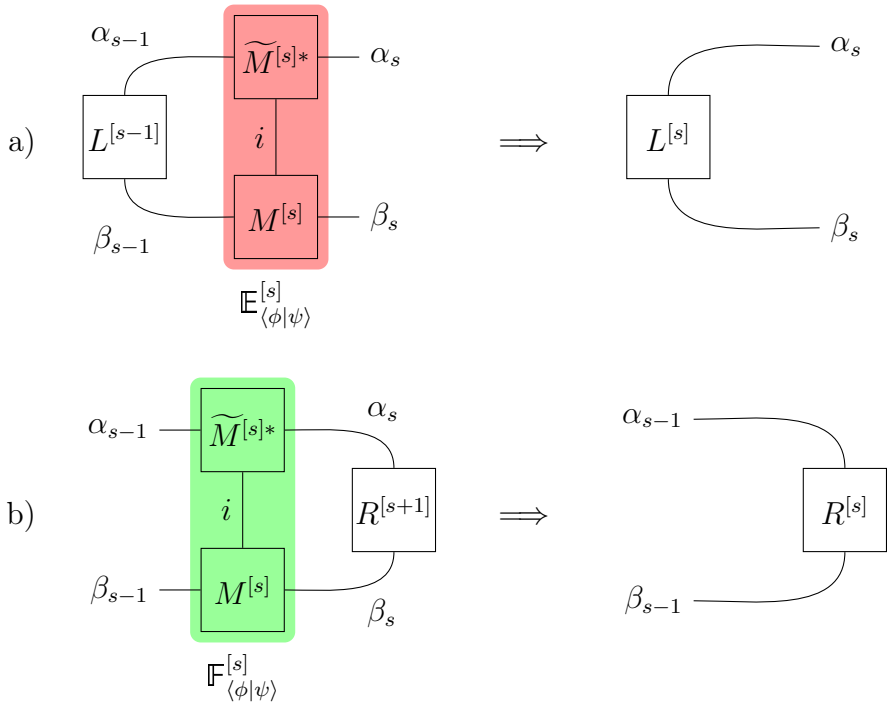


Figure 2.6: Graphical representation of equation (2.95) for a) and equation (2.98) for b); the transfer operators are highlighted. This process of fusing the matrices of the MPS together is what, actually, reduces the complexity of the problem from exponential to linear in the number of sites.

$$\mathbb{E}^{[s]}_{\langle\phi|\psi\rangle} := \sum_{i=1}^{d} \widetilde{M}_i^{[s]*} \otimes M_i^{[s]}; \tag{2.94}$$

that does exactly what we expect:

$$\mathbb{E}^{[s]}_{\langle\phi|\psi\rangle}(L^{[s-1]}) = \sum_{i=1}^{d} \widetilde{M}_i^{[s]\dagger} L^{[s-1]} M_i^{[s]} = L^{[s]}. \tag{2.95}$$

When it is obvious from the context, we drop the subscript that indicates it is a transfer operator for overlaps. There are other kinds of transfer operators; in this work, we will define transfer operators in the next subsection and in section 2.12. Despite their cumbersome aspect, they are just simple slices of overlap's tensor network (see figure 2.6a).

What we have just defined is the transfer operator that computes the overlap and the matrices $L^{[s]}$ from left to right. For the $\mathbb{F}^{[s]}$'s that compute from right to left, the definition is basically the same as equation (2.95). To appreciate the difference between the two operators we have to go deeper, to

the level of coefficients. Think about $L^{[s]}$ not as a matrix but as a single long vector; then, $\mathbb{E}^{[s]}$ becomes a matrix and its action is a matrix multiplication. The coefficients of $\mathbb{E}^{[s]}$ would be:

$$\left(\mathbb{E}^{[s]}\right)_{(\alpha_s\beta_s),(\alpha_{s-1}\beta_{s-1})} = \sum_{i=1}^{d} (\widetilde{M}_i^{[s]})^*_{\alpha_{s-1}\alpha_s} (M_i^{[s]})_{\beta_{s-1}\beta_s}. \tag{2.96}$$

A simple verification proves that (2.95) is correct (see again figure 2.6a). For the transfer operators that goes backwards we have the coefficients:

$$\left(\mathbb{F}^{[s]}\right)_{(\alpha_{s-1}\beta_{s-1}),(\alpha_s\beta_s)} = \sum_{i=1}^{d} (M_i^{[s]})_{\beta_{s-1}\beta_s} (\widetilde{M}_i^{[s]})^*_{\alpha_{s-1}\alpha_s}. \tag{2.97}$$

from which we properly obtain the $R^{[s]}$ matrices:

$$\mathbb{F}^{[s]}(R^{[s]}) = \sum_{i=1}^{d} M_i^{[s]} R^{[s]} \widetilde{M}_i^{[s]\dagger} = R^{[s-1]}, \tag{2.98}$$

see figure 2.6b.

When we speak about complexities, it suffices to specify only the cost and the optimal contraction to compute one transfer operator. Then, if we need the complexity for all the chain and all the operations are the same, we just multiply it by the number of sites. For both $\mathbb{E}^{[s]}$ and $\mathbb{F}^{[s]}$, the complexity of one occurrence is:

$$\mathcal{O}(d\widetilde{D}_s\widetilde{D}_{s-1}D_{s-1}) + \mathcal{O}(d\widetilde{D}_s D_{s-1}D_s). \tag{2.99}$$

From equation (2.92), we learn that, when we deal with the same state and the matrices are left- or right-normalized, important simplifications are possible. Indeed, whenever site s is left-normalized, it is true that $\mathbb{E}^{[s]}(\mathbb{1}) = \mathbb{1}$ and, each time site t is right-normalized, it holds $\mathbb{F}^{[t]}(\mathbb{1}) = \mathbb{1}$. Hence, if all the sites from 1 to s are left-normalized, we have $L^{[s']} = \mathbb{1}$, $\forall s' = 1, \ldots, s$. Moreover, if all the sites from t to N are right-normalized, we obtain $R^{[t']} = \mathbb{1}$, $\forall t' = t, \ldots, N$. We will see applications of these results in the next subsection.

2.7.3 Expectation values

In this subsection, we only consider operators that can be expressed as the simultaneous action of single-site operators (see also section 2.6) on multiple sites. These operators are products $O^{[i]}O^{[j]}\cdots O^{[n]}$, that act on sites $i, j, \ldots, n$, and where:

$$O^{[s]} = \sum_{i,j=1}^{d} (O^{[s]})_{ij} \left|i\right\rangle\!\left\langle j\right|. \tag{2.100}$$

In the most general case, we could have an operator for every site and, then, we could allow the possibility for some of those operators to be just identities. We consider this kinds of operators because, for them, the situation is simpler (for expectation values involving the most generic operators possible see section 2.12). The optimal contractions for the matrix elements $\langle\phi|O^{[1]}O^{[2]}\cdots O^{[N]}|\psi\rangle$ are similar to equation (2.88), except that the single sums are now run over two physical indices and operators have entered:

$$\begin{aligned}\langle\phi|O^{[1]}O^{[2]}\cdots O^{[N]}|\psi\rangle = \\ = \sum_{i_N,j_N=1}^{d}(O^{[N]})_{i_N j_N}\widetilde{M}_{i_N}^{[N]\dagger}\left(\cdots\left(\sum_{i_1,j_1=1}^{d}(O^{[1]})_{i_1 j_1}\widetilde{M}_{i_1}^{[1]\dagger}M_{i_1}^{[1]}\right)\cdots\right)M_{i_N}^{[N]};\end{aligned} \tag{2.101}$$

with analogous reversion steps as for the overlaps, if we want to start from the right hand side of the chain.

For the physical problems we are interested in, we often need expectation values of operators that act on no more than two sites at once. Consider the operator $O^{[s]}$ that acts on site s and that could be, for example, the magnetization along a particular axis for the spin on that site. The tensor network that represents $\langle\psi|O^{[s]}|\psi\rangle$ is in figure 2.7. In this situation, we have $N-1$ steps similar to the overlap case and one slightly more complex step for site s. We define the new special transfer operator (for matrix elements) as:

$$\mathbb{E}^{[s]}_{\langle\phi|O^{[s]}|\psi\rangle} \coloneqq \sum_{i,j=1}^{d}(O^{[s]})_{ij}\widetilde{M}_i^{[s]*}\otimes M_j^{[s]}. \tag{2.102}$$

We will refer to it also as simply $\mathbb{E}^{[s]}_{O^{[s]}}$. The optimal order of contraction, for a single application, must be made explicit with the coefficients:

$$\begin{aligned}(L^{[s]})_{\alpha_s\beta_s} = \sum_{\alpha_{s-1}=1}^{\widetilde{D}_{s-1}}\sum_{j=1}^{d}(\widetilde{M}_i^{[s]})^*_{\alpha_{s-1}\alpha_s} \\ \times\left(\sum_{j=1}^{d}(O^{[s]})_{ij}\left(\sum_{\beta_{s-1}=1}^{D_{s-1}}(L^{[s-1]})_{\alpha_{s-1}\beta_{s-1}}(M_i^{[s]})^*_{\beta_{s-1}\beta_s}\right)\right);\end{aligned} \tag{2.103}$$

this has a cost of $\mathcal{O}(dD_sD_{s-1}\widetilde{D}_{s-1})+\mathcal{O}(d^2D_s\widetilde{D}_{s-1})+\mathcal{O}(dD_s\widetilde{D}_s\widetilde{D}_{s-1})$.

Suppose, now, that $|\psi\rangle$ is, actually, in mixed-canonical form, as in equation (2.29), but with the unknown normalization on site s instead of $s+1$. We have that all the $L^{[s']}$, with $s'<s$, and all the $R^{[t']}$, with $t'>s$, are

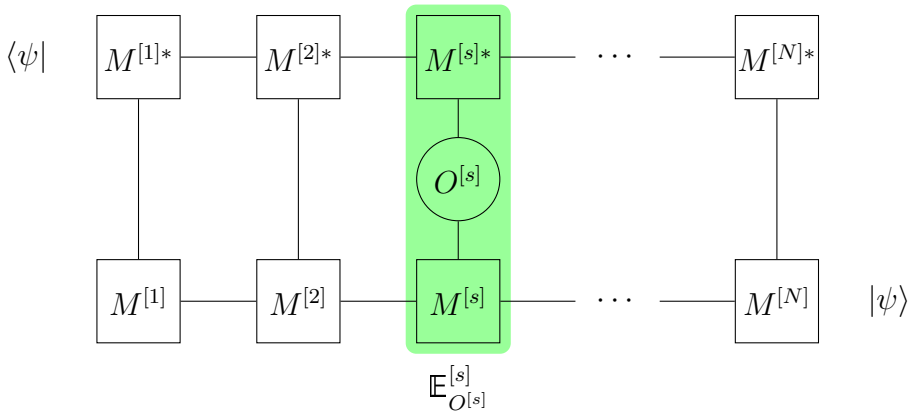


Figure 2.7: Tensor network for $\langle\psi|O^{[s]}|\psi\rangle$, with highlighted new transfer operator for expectation values.

equal to the identity. The tensor network of the expectation $\langle\psi|O^{[s]}|\psi\rangle$ is drastically reduced (see figure 2.8):

$$\langle\psi|O^{[s]}|\psi\rangle = \sum_{i,j=1}^{d} (O^{[s]})_{ij} \operatorname{Tr}\Big[M_j^{[s]\dagger} M_i^{[s]} \Big] . \tag{2.104}$$

Analogous considerations can be made for operators acting on two sites. At this point, it is fairly easy to imagine what will happen to the expectation value $\langle\psi|O^{[s]}O^{[t]}|\psi\rangle$, when $|\psi\rangle$ is left-normalized from 1 to $s-1$ and right-normalized from $t+1$ to N:

$$\langle\psi|O^{[s]}O^{[t]}|\psi\rangle = \left(\mathbb{E}_{O^{[s]}}^{[s]} \circ \mathbb{E}^{[s+1]} \circ \cdots \circ \mathbb{E}^{[t-1]} \circ \mathbb{E}_{O^{[t]}}^{[t]} \right) (\mathbb{1}). \tag{2.105}$$

The transfer operators without subscript are those of the overlap.

2.8 MPS sums

A fundamental operation with vectors is the sum, but it is not very popular among MPS algorithms. The reason is practical: a sum increases the bond link and thus the computational cost of further operations. This often requires a compression afterwards, yet another costly operation.

Let us have a look at how it works; for once it is easier to start from two PBC MPS, namely:

$$|\psi\rangle = \sum_{i_1,\ldots,i_N} \operatorname{Tr}\Big[M_{i_1}^{[1]} \cdots M_{i_N}^{[N]} \Big] |i_1 \cdots i_N\rangle \tag{2.106}$$

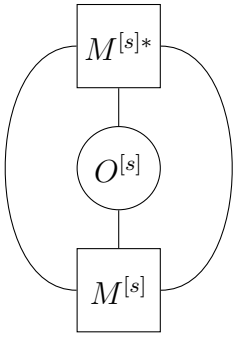


Figure 2.8: What remains of the expectation value $\langle\psi|O^{[s]}|\psi\rangle$, when $|\psi\rangle$ is in the proper mixed-canonical form; confront with equation (2.104)

$$|\phi\rangle = \sum_{i_1,\ldots,i_N} \mathrm{Tr}\Big[\, \widetilde{M}^{[1]}_{i_1} \cdots \widetilde{M}^{[N]}_{i_N} \Big]\, |i_1 \cdots i_N\rangle\,; \tag{2.107}$$

the sum of the two vectors implies the sum of the coefficients and we just have to remember that the sum of two traces is the trace of the direct sum, $\mathrm{Tr}[\,A\,] + \mathrm{Tr}[\,B\,] = \mathrm{Tr}[\,A \oplus B\,]$:

$$\begin{aligned} \mathrm{Tr}\Big[\, M^{[1]}_{i_1} \cdots M^{[N]}_{i_N} \Big] + \mathrm{Tr}\Big[\, \widetilde{M}^{[1]}_{i_1} \cdots \widetilde{M}^{[N]}_{i_N} \Big] = \\ = \mathrm{Tr}\left[\begin{pmatrix} M^{[1]}_{i_1} \cdots M^{[N]}_{i_N} & 0 \\ 0 & \widetilde{M}^{[1]}_{i_1} \cdots \widetilde{M}^{[N]}_{i_N} \end{pmatrix} \right] = \\ = \mathrm{Tr}\left[\begin{pmatrix} M^{[1]}_{i_1} & 0 \\ 0 & \widetilde{M}^{[1]}_{i_1} \end{pmatrix} \cdots \begin{pmatrix} M^{[N]}_{i_N} & 0 \\ 0 & \widetilde{M}^{[N]}_{i_N} \end{pmatrix} \right]. \end{aligned} \tag{2.108}$$

Clearly the MPS form is preserved and we have found how to produce the MPS sum matrices, i.e., glueing together the addends' original matrices with a direct sum, for every site and matrix:

$$N^{[s]}_i = M^{[s]}_i \oplus \widetilde{M}^{[s]}_i = \begin{pmatrix} M^{[s]}_i & 0 \\ 0 & \widetilde{M}^{[s]}_i \end{pmatrix} \quad \forall s, \forall i; \tag{2.109}$$

this is why the bond link grows: if matrix dimensions initially were $D_{s-1} \times D_s$ and $\widetilde{D}_{s-1} \times \widetilde{D}_s$, now they are $(D_{s-1} + \widetilde{D}_{s-1}) \times (D_s + \widetilde{D}_s)$.

The OBC case is very similar: the coefficients are scalars and they are equal to their traces, thus we can start in the same way as before. However, we end up with the trace of a 2×2 matrix, not a scalar, and the OBC form is lost. To recover it, we use a simple trick that consists in writing first and

last site matrices as:

$$N_i^{[1]} = \begin{pmatrix} M_i^{[1]} & \widetilde{M}_i^{[1]} \end{pmatrix}, \qquad N_i^{[N]} = \begin{pmatrix} M_i^{[N]} \\ \widetilde{M}_i^{[N]} \end{pmatrix}; \tag{2.110}$$

a simple verification proves that it works.

A small remark on the augment of bond link: sometimes we do not fully need the new $D+\widetilde{D}$ dimensions. The limit case is the sum of the same vector, $|\psi\rangle+|\psi\rangle = 2\,|\psi\rangle$: the bond link doubles but we know that the resulting vector should just have the coefficients rescaled. The increase in the bond link is sometimes *apparent* and it is good practice to subsequently compress the result of a sum, maybe checking the Schmidt coefficients (see section 2.3.3) that reveal the optimal bond link dimension.

2.9 MPS compression

The necessity of a compression scheme rises from the existence of certain operations, such as sums or MPO·MPS products, that increase the dimension of the bond link. In order to simplify the MPS notation, we divide the set of all MPS into classes label by the maximum bond link dimension of the MPS in said class. We need a method that, taken an MPS of class $\widetilde{D}$, returns another MPS of class $D \leq \widetilde{D}$ that is the best approximation of the original one that we can afford[2].

There are basically two main roads to compress an MPS: the first one exploits the low-rank approximation of SVD (see theorem A.1). The other one iteratively updates every matrix of a new, smaller MPS until the closest MPS is found; this is a variational approach. Both these methods have advantages and disadvantages.

2.9.1 SVD compression

There is a small preamble to be made; take the bipartite state $|\widetilde{\psi}\rangle$:

$$|\widetilde{\psi}\rangle = \sum_{i=1}^{m}\sum_{j=1}^{n} \widetilde{\psi}_{ij} |i\rangle_A |j\rangle_B. \tag{2.111}$$

We can identify the coefficients $\widetilde{\psi}_{ij}$ as the elements of a $m \times n$ matrix $\widetilde{\Psi}$. A remarkable fact is that the 2-norm of $|\widetilde{\psi}\rangle$ coincides with the *Frobenius norm*

[2]In the rest of this work, we sometimes refer to the class of a certain MPS ...

of $\widetilde{\Psi}$:

$$\left\||\widetilde{\psi}\rangle\right\|_2^2 = \sum_{i,j} \left|\widetilde{\psi}_{ij}\right|^2 = \left\|\widetilde{\Psi}\right\|_F. \tag{2.112}$$

From theorem A.1, we know that $\widetilde{\Psi} = \widetilde{U}\widetilde{\Sigma}\widetilde{V}^\dagger$ (the SVD) is optimally approximated in Frobenius norm by $\Psi = U\Sigma V^\dagger$, if Σ contains the greatest singular values of $\widetilde{\Sigma}$ and U and V contain the corresponding left and right singular vectors from $\widetilde{U}$ and $\widetilde{V}$ respectively.

State $|\widetilde{\psi}\rangle$ can be written in Schmidt form too:

$$|\widetilde{\psi}\rangle = \sum_{k=1}^{\widetilde{D}} \sigma_k |e_k\rangle_A |f_k\rangle_B, \tag{2.113}$$

where $\widetilde{D}$ is the number of non-zero Schmidt coefficients. Remembering how the Schmidt decomposition is constructed from the SVD of $\widetilde{\Psi}$ (section 1.3), it is easy to understand why the optimal approximation of $|\widetilde{\psi}\rangle$ is given by the vector:

$$|\psi\rangle = \sum_{k=1}^{D} \sigma_k |e_k\rangle_A |f_k\rangle_B, \tag{2.114}$$

i.e., supposing the σ_k are in decreasing order, retaining the first $D \leq \widetilde{D}$ Schmidt coefficients (singular values) and the first D basis vector for A and B (associated left and right singular vectors).

Going back to the compression of MPSs, we saw in section 2.3.3 that, from the mixed-canonical form with orthonormality center at bond n, we can read directly the Schmidt decomposition for bipartition $1 \ldots n : n+1 \ldots N$:

$$|\widetilde{\psi}\rangle = \sum_{i_1,\ldots,i_N=1}^{d} \widetilde{A}^{[1]}_{i_1} \cdots \widetilde{A}^{[n]}_{i_n} \widetilde{\Sigma}^{[n]} \widetilde{B}^{[n+1]}_{i_{n+1}} \cdots \widetilde{B}^{[N]}_{i_N} |i_1 \cdots i_N\rangle\,, \tag{2.115}$$

then, if the bond link has dimension $\widetilde{D}$, we can shrink it right away, with maximum precision, retaining only the first $D \leq \widetilde{D}$ columns of $\widetilde{A}^{[n]}_{i_n}$, the first D rows of $\widetilde{B}^{[n+1]}_{i_{n+1}}$ and the first D diagonal elements of $\widetilde{\Sigma}^{[n]}$. It is only possible to do this at an orthonormality center, thus, in order to shrink every bond link, we have to sweep through every possible mixed-canonical form.

Suppose we have to reduce bond n and we are moving from left to right, then the algorithm could be somewhat like this:

Algorithm 2.1: SVD compression

1. follow the procedure of left normalization for site n (as described in section 2.3.1), but stop before multiplying $\widetilde{V}^{[n]\dagger}$ to the right:

$$|\widetilde{\psi}\rangle = \sum \cdots A^{[n-1]} \widetilde{A}^{[n]} \widetilde{\Sigma}^{[n]} \widetilde{V}^{[n]\dagger} \widetilde{B}^{[n+1]} \cdots ; \tag{2.116}$$

2. truncate matrices $\widetilde{A}^{[n]}_{i_n}$, $\widetilde{\Sigma}^{[n]}$ and $\widetilde{V}^{[n]\dagger}$ to a maximum of, respectively, D columns, D diagonal elements and D rows; we call these smaller matrices $A^{[n]}_{i_n}$, $\Sigma^{[n]}$ and $V^{[n]\dagger}$:

$$|\widetilde{\psi}\rangle = \sum \cdots A^{[n-1]} A^{[n]} \Sigma^{[n]} V^{[n]\dagger} \widetilde{B}^{[n+1]} \cdots ; \tag{2.117}$$

3. multiply $\Sigma^{[n]} V^{[n]\dagger}$ to the right and shift to the next bond on the right.

This procedure has to be applied from bond 1 to bond $N-1$ and, at the beginning, the MPS must have orthonormality center at bond 1, otherwise a (costly) canonization process must be applied[3].

The advantages of SVD compression are that it is fast and convenient for small compressions and it is quite easy to implement on a machine. The disadvantage is that it is not optimal: although for a single bond it is mathematically the best approximation, the truncation errors tend to accumulate during the sweep and the final result is not always reliable. Anyway, if the truncation errors are small (say compared to the norm of the vector), we can trust the smaller MPS.

Since we will need it later, let us analyse the complexity of the SVD compression. The most expensive step of the algorithm 2.1 is the SVD calculated at point 1. This is performed on a matrix of dimensions $dD \times \widetilde{D}$ (the vertical merger of the matrices of the site). The cost of SVD for a matrix $m \times n$ is $\mathcal{O}(mn^2)$ if $m \geq n$, then the algorithm cost (for every site) is $\mathcal{O}(dD\widetilde{D}^2)$ if $dD \geq \widetilde{D}$ or $\mathcal{O}(d^2D^2\widetilde{D})$ if $\widetilde{D} \geq dD$. If the latter is the case, we must add the cost of the multiplication of $V^{[n]\dagger}$ on the right: $\mathcal{O}(dD\widetilde{D}^2)$. As already mentioned, if $\widetilde{D} \gg D$ then SVD compression should be avoided because it becomes slower than variational compression. Finally, if the canonization process needs to be applied before the algorithm, the total cost becomes

[3]Obviously we may decide to go backwards from bond $N-1$ to 1 applying right normalizations and with the orthonormality center initially at bond $N-1$, there is no difference.

$\mathcal{O}(NdD^3)$, very expensive.

2.9.2 Variational compression

The formally correct way to compress MPS $|\widetilde{\psi}\rangle$, of class $\widetilde{D}$, is to run through all the MPSs $|\psi\rangle$ of class D ($< \widetilde{D}$) and find the one that minimizes:

$$\big\| |\widetilde{\psi}\rangle - |\psi\rangle \big\|_2^2 = \langle\widetilde{\psi}|\widetilde{\psi}\rangle - \langle\psi|\widetilde{\psi}\rangle - \langle\widetilde{\psi}|\psi\rangle + \langle\psi|\psi\rangle\,. \tag{2.118}$$

We treat the matrices M of $|\psi\rangle$ as parameters (more precisely, the elements of the matrices are the parameters), but it is very difficult to minimize the distance with respect to all these variables. Instead, we can fix the value of all matrices except those of one site, say the n-th, and find what matrices minimize the reduced problem. Afterwards, we move to another site, we fix the matrices that are not on that site and we find the matrices that, for that site, minimize the distance and so on. This is a variational approach: at every step the distance does not increase, so, sweeping through the chain many times, we hope to find the MPS with the minimum possible distance.

To find $M^{[n]}_{i_n}$ that extremize the distance, we put equal to zero the derivative with respect to $(M^{[n]}_{i_n})^*_{\alpha_{n-1}\alpha_n}$, which is found only in $\langle\psi|\widetilde{\psi}\rangle$ and $\langle\psi|\psi\rangle$:

$$\begin{aligned}
&\frac{\partial}{\partial\Big((M^{[n]}_{i_n})^*_{\alpha_{n-1}\alpha_n}\Big)}(\langle\psi|\psi\rangle - \langle\psi|\widetilde{\psi}\rangle) = \\
&\qquad \sum_{i_1,\ldots,\hat{i}_n,\ldots,i_N=1}^{d} (M^{[1]*}_{i_1}\cdots M^{[n-1]*}_{i_{n-1}})_{1,\alpha_{n-1}} (M^{[n+1]*}_{i_{n+1}}\cdots M^{[N]*}_{i_N})_{\alpha_n,1} \\
&\qquad\qquad \times M^{[1]}_{i_1}\cdots M^{[n]}_{i_n}\cdots M^{[N]}_{i_N} + \\
&- \sum_{i_1,\ldots,\hat{i}_n,\ldots,i_N=1}^{d} (M^{[1]*}_{i_1}\cdots M^{[n-1]*}_{i_{n-1}})_{1,\alpha_{n-1}} (M^{[n+1]*}_{i_{n+1}}\cdots M^{[N]*}_{i_N})_{\alpha_n,1} \\
&\qquad\qquad \times \widetilde{M}^{[1]}_{i_1}\cdots \widetilde{M}^{[n]}_{i_n}\cdots \widetilde{M}^{[N]}_{i_N} = 0,
\end{aligned} \tag{2.119}$$

where the hat over i_n indicates that the sum over that index is skipped. Look at the meaning of this complicated expression in the graphical representation of figure 2.9, where we indicated the bigger bond link of $|\widetilde{\psi}\rangle$ with a double line.

If we suppose that $|\psi\rangle$ is in canonical form, then the expression (and the graph) is much simpler and we can actually read off directly the new value of $M^{[n]}_{i_n}$ that will be substituted. The exact position of the orthonormality

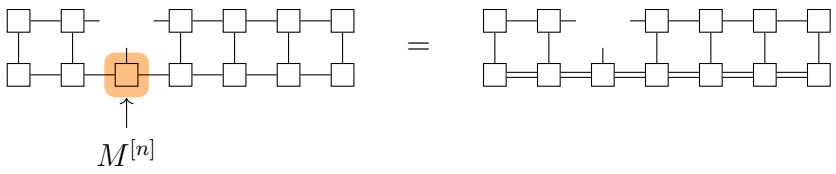


Figure 2.9: The derivative of $\langle\psi|\psi\rangle$ on the left is equal to the derivative of $\langle\psi|\widetilde{\psi}\rangle$ on the right; it is the graphical representation of equation (2.119). The matrix we want to know the value of is highlighted and we used a double line to indicate the larger bond link of $|\widetilde{\psi}\rangle$.

center is not important, it is sufficient to have this form:

$$|\psi\rangle = \sum_{i_1,\dots,i_N=1}^{d} A^{[1]}_{i_1} \cdots A^{[n-1]}_{i_{n-1}} M^{[n]}_{i_n} B^{[n+1]}_{i_{n+1}} \cdots B^{[N]}_{i_N} |i_1 \cdots i_N\rangle \,. \tag{2.120}$$

In this way, all the transfer operators on the right and on the left of $M^{[n]}$ produce identity matrices. To simplify the equations, let us define block L and block R:

$$\begin{aligned} L^{[n-1]} &= \sum_{i_{n-1}=1}^{d} A^{[n-1]\dagger}_{i_{n-1}} \left(\cdots \left(\sum_{i_1=1}^{d} A^{[1]\dagger}_{i_1} \widetilde{M}^{[1]}_{i_1} \right) \cdots \right) \widetilde{M}^{[n-1]}_{i_{n-1}}, \\ R^{[n+1]} &= \sum_{i_{n+1}=1}^{d} \widetilde{M}^{[n+1]}_{i_{n+1}} \left(\cdots \left(\sum_{i_N=1}^{d} \widetilde{M}^{[N]}_{i_N} B^{[N]\dagger}_{i_N} \right) \cdots \right) B^{[n+1]\dagger}_{i_{n+1}}; \end{aligned} \tag{2.121}$$

block $L^{[n-1]}$ is the application of the left transfer operators of $\langle\psi|\widetilde{\psi}\rangle$ from site 1 to site $n-1$; similarly, $R^{[n+1]}$ is formed by the right transfer operators from site N to site $n+1$. Figure 2.10 illustrates how the network of tensors of equation (2.119) is simplified after these assumptions and it represents the following direct computation of the optimal matrices:

$$M^{[n]}_{i_n} = L^{[n-1]} \widetilde{M}^{[n]}_{i_n} R^{[n+1]}, \quad \forall i_n = 1, \dots, d. \tag{2.122}$$

Have a look at the algorithm that systematically applies this process to the whole MPS:

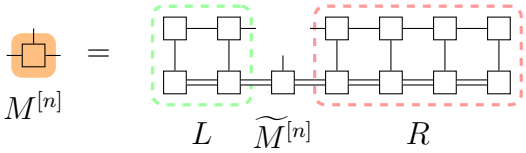


Figure 2.10: The simplified tensor network that computes the improved matrices for site n; it represents equation (2.122)

Algorithm 2.2: Variational compression

1. set the `direction` of the sweep either left to right (`LtoR`) or right to left (`RtoL`);
2. obtain initial guess of $|\psi\rangle$, e.g. randomly;
3. normalize the matrices of $|\psi\rangle$ such that it is in the mixed-canonical form of equation (2.120) with center at site n;
4. create blocks $L^{[n-1]}$ and $R^{[n+1]}$ from equation (2.121), *saving the partial results for every site* (i.e. $L^{[n-2]}$, $R^{[n+2]}$, $L^{[n-3]}$, ...);
5. $M^{[n]}_{i_n} \leftarrow L^{[n-1]} \widetilde{M}^{[n]}_{i_n} R^{[n+1]}, \quad \forall i_n = 1, \ldots, d$;
6. if $n = 1$ or $n = N$ invert the value of `direction`;
7. if `direction` is `LtoR` then:
   (a) shift the center of the mixed-canonical form to the right;
   (b) regenerate block $L^{[n]}$ from $L^{[n-1]}$ because matrices $M^{[n]}$ now have changed;
   (c) leave alone block $R^{[n+2]}$ (saved before) because no matrices that previously generated it have changed so far;
   (d) restart from point 5 with $n \leftarrow n + 1$;

   otherwise:
   (a) shift the center of the mixed-canonical form to the left;
   (b) regenerate block $R^{[n]}$ from $R^{[n+1]}$ because matrices $M^{[n]}$ now have changed;
   (c) leave alone block $L^{[n-2]}$ (saved before) because no matrices that previously generated it have changed so far;
   (d) restart from point 5 with $n \leftarrow n - 1$.

To probe the convergence, we have to estimate the distance between the two vectors. Computing naively equation (2.118) at every step would incur in a great waste of resources. Instead, we note from figure 2.10 that both $\langle\psi|\widetilde{\psi}\rangle$ and $\langle\widetilde{\psi}|\psi\rangle$ are equal to:

$$\sum_{i_n=1}^{d} \mathrm{Tr}\Big[\, M_{i_n}^{[n]\dagger} M_{i_n}^{[n]} \,\Big]\,; \tag{2.123}$$

but, given the mixed-canonical form of $|\psi\rangle$, this is also the value of $\langle\psi|\psi\rangle$. In conclusion, we have:

$$\big\| |\widetilde{\psi}\rangle - |\psi\rangle \big\|_2^2 = \widetilde{\nu}^2 - \sum_{i_n=1}^{d} \mathrm{Tr}\Big[\, M_{i_n}^{[n]\dagger} M_{i_n}^{[n]} \,\Big]\,, \tag{2.124}$$

where $\widetilde{\nu} = \big\| |\widetilde{\psi}\rangle \big\|_2$ is calculated only once at the beginning of the compression.

The variational compression has several benefits: first of all, it is optimal. Then, its speed can be greatly adjusted changing the setting that assesses convergence. In particular, if we are satisfied with a rough compression, we can spare a lot of sweeps (and time). This is not the case for SVD compression, where it is not possible to decide the quality of the approximation. Moreover, when the increase in bond link is only apparent (as it happens for MPS sums), the convergence is faster too. A fourth advantage is the greater speed for large compressions: we determined empirically that, when $\widetilde{D}$ is 4 or 5 times D, then variational compression is faster than SVD compression. This leads to the conclusion that when MPS sums are involved, the better choice is SVD compression, while with MPO products we will see that the increased bond link needs a variational compression.

Variational compression has also some minor drawbacks, the worst being its need for a good initial guess. If the initial vector is chosen randomly, a lot of time is wasted setting its matrices properly. Instead, feeding the algorithm with the result of a SVD compression results in a much faster conversion. This technique could be used when a lot of precision is required. Another downside is that it is slightly harder to write an implementation for it. Finally, as for any variational method, there is the danger to get stuck in a local minimum (because we are not changing the matrices globally, but only one by one). To deal with this problem, we could change the matrices of two sites simultaneously. This has a greater cost but the details of it will not be discussed here.

It is very difficult to evaluate the complexity of variational compression, because it is *a priori* unclear how the total number of sweeps (that we call χ_v)

depends on various factors. For example, it surely depends on the precision required, but we could not find the exact relation. Moreover, we do not know if the number of sites N or the bond link dimensions D and $\widetilde{D}$ influence somehow χ_v. However, for our choice of parameters and for the cases we monitored, we have found χ_v to be always a number in the order of hundreds and thus we consider it as a constant factor (that we report explicitly in the orders of complexity). The main loop of the algorithm has three steps: the double multiplication and growth of block L (or R), that both have complexity $\mathcal{O}(dD\widetilde{D}^2)+\mathcal{O}(dD^2\widetilde{D})$, plus the shift of the orthonormality center toward left or right, that has a cost of $\mathcal{O}(dD^3)$. The total complexity is:

$$\mathcal{O}(\chi_v NdD(\widetilde{D}^2 + D\widetilde{D} + D^2)) \approx \mathcal{O}(\chi_v NdD(D+\widetilde{D})^2); \tag{2.125}$$

again, since we do not know if χ_v depends on N, D, $\widetilde{D}$, d or else, we cannot be sure of, for example, the linear growth of the complexity in N, but we did not observed any significant evidence of that either.

2.10 Matrix Product Operators

From this section, we will talk about the generalization of the matrix product states to operators, called matrix product operators. Here, we introduce them and in the next section we will explain products with MPS and other operators. In the section after, we will see how to compute transfer operators and expectation values. Finally, we present the procedure to write a generic many-body hamiltonian as a matrix product operator.

The most generic operator O defined for our system on a lattice has the following form:

$$O = \sum_{\substack{i_1,\ldots,i_N=1 \\ j_1,\ldots,j_N=1}}^{d} f_{j_1\cdots j_N}^{i_1\cdots i_N} \left|i_1\cdots i_N\right\rangle\!\left\langle j_1\cdots j_N\right|. \tag{2.126}$$

These operators do not guarantee that the MPS form is preserved. To overcome the problem, the formalism of *Matrix Product Operators* (MPO) was introduced. MPOs emerge quite naturally from the observation that MPS have only one physical index per site and they are vectors, so we could add a second physical index to render them operators. Indeed the most general MPO is:

$$O = \sum_{\substack{i_1,\ldots,i_N=1 \\ j_1,\ldots,j_N=1}}^{d} W_{i_1 j_1}^{[1]} W_{i_2 j_2}^{[2]} \cdots W_{i_N j_N}^{[N]} \left|i_1\cdots i_N\right\rangle\!\left\langle j_1\cdots j_N\right|. \tag{2.127}$$

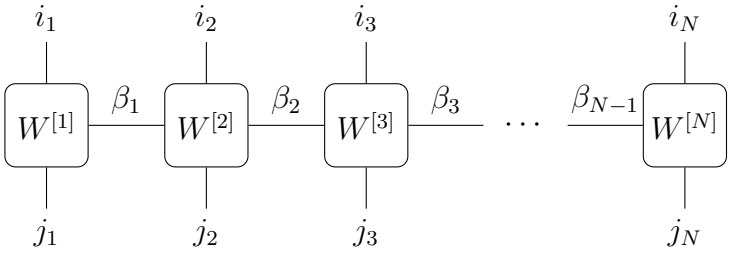


Figure 2.11: Graphical representation of a matrix product operator (with OBC).

MPOs possess a bond link dimension too, we distinguish it from other bond dimensions around adding the name of the operator or the name of its matrices as a subscript: D_W.

We give a graphical representation for MPOs too: they are an hybrid between MPSs (with square shape) and operators (with circular shape) so we depict their sites as squares with rounded corners. Every sites has four legs: two for the virtual indices along the horizontal direction (as for MPS) and two for the physical indices along the vertical direction (as for single site operators); see figure 2.11.

There are many procedures designed for MPSs that do not change for MPOs. It is sufficient to regroup the indices (i_s, j_s) as a single index, in this way the MPO is reduced to an MPS with a larger local Hilbert space of dimension d^2. For example, not surprisingly, every operator like (2.126) can be written in MPO form. The proof is the same as given in section 2.5: regrouping the corresponding indices (i_s, j_s) of f, we obtain the coefficients c of a generic vector. This comes, again, at the cost of an exponentially increasing bond link in the middle of the chain. Other algorithms, that are easily promoted for MPOs, are the sum (section 2.8) and the compressions (section 2.9). The only new operations we will need are the products.

2.11 MPO products

There are two kinds of products: MPO·MPS products or MPO·MPO products; they both have the remarkable property of preserving the MPS/MPO structure. The cost is a sensible augment of the bond link dimension of the result.

Consider an MPO O acting on an MPS $|\psi\rangle$:

$$O|\psi\rangle = \sum_{\substack{i_1,\ldots,i_N=1 \\ j_1,\ldots,j_N=1}}^{d} W^{[1]}_{i_1 j_1} \cdots W^{[N]}_{i_N j_N} M^{[1]}_{j_1} \cdots M^{[N]}_{j_N} |i_1 \cdots i_N\rangle \tag{2.128}$$

$$= \sum_{i_1,\ldots,i_N=1}^{d} \cdots \left(\sum_{j_s=1}^{d} W^{[s]}_{i_s j_s} \otimes M^{[s]}_{j_s} \right) \cdots |i_1 \cdots i_N\rangle \,, \tag{2.129}$$

in the second line the multiplication of the two strings is converted in the (equivalent) tensor product of them. The latter is, in turns, splitted in the single sites and every term incorporates its local sum of j_s. For every site, this results in a new 3-tensor $Q^{[s]}_i$ that, in terms of the matrix elements, is:

$$(Q^{[s]}_i)_{(\beta,\alpha),(\beta',\alpha')} = \sum_{j=1}^{d} (W^{[s]}_{ij})_{\beta\beta'} (M^{[s]}_j)_{\alpha\alpha'}, \tag{2.130}$$

this is represented in figure 2.12 a). We see from the tensor product that the result have bond dimension $D_W D$ and most certainly it has to be compressed somehow. The total operation has a number of operations of:

$$\mathcal{O}(N d^2 D_W^2 D^2). \tag{2.131}$$

In a similar fashion, MPO·MPO products produce, for every site, a new 4-tensor $Z^{[s]}_{ij}$:

$$(Z^{[s]}_{ij})_{(\beta,\alpha),(\beta',\alpha')} = \sum_{k=1}^{d} (W^{[s]}_{ik})_{\beta\beta'} (T^{[s]}_{kj})_{\alpha\alpha'}, \tag{2.132}$$

it is depicted in figure 2.12 b); its complexity is:

$$\mathcal{O}(N d^3 D_W^2 D_T^2). \tag{2.133}$$

2.12 Transfer operators and expectation values for MPOs

Matrix product operators come with their own transfer operators for the evaluation of expectation values and matrix elements. However, in this case, the left and right blocks obtained are not simple matrices, but 3-tensors: we must account for the MPO bond index.

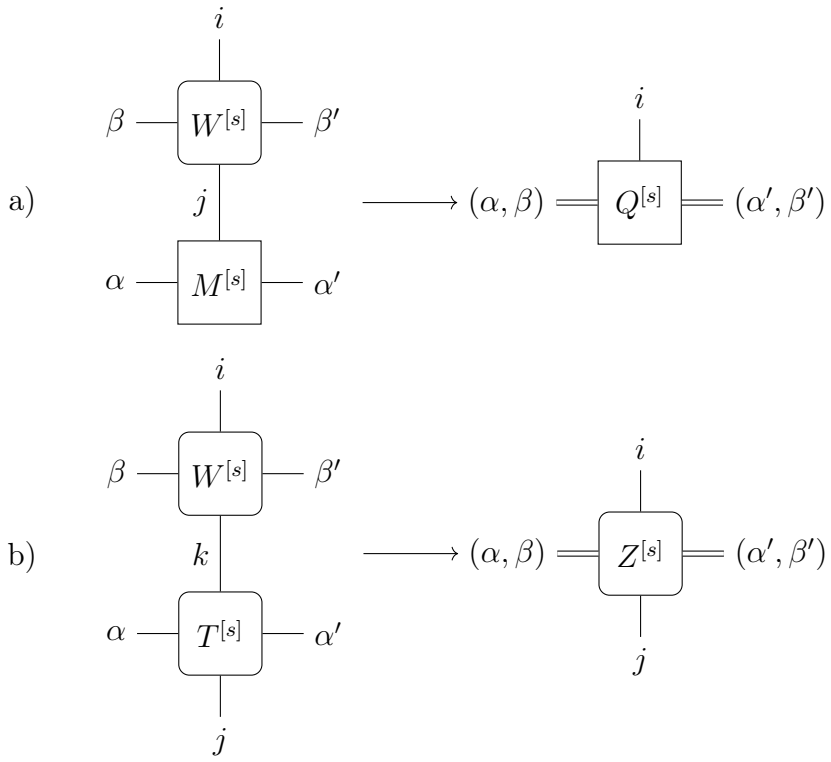


Figure 2.12: Graphical representation of MPO products for single sites: a) MPO·MPS product (equation (2.130)), b) MPO·MPO product (equation (2.132)). Again, the double line indicates a bigger bond link.

We start defining the first 3-tensor of the matrix element $\langle\phi|O|\psi\rangle$ (see figure 2.13) by contracting the physical indices of the first site matrices:

$$\left(L^{[1]}\right)_{\alpha'_1\beta_1\alpha_1} = \sum_{i=1}^{d} \left(\widetilde{M}_i^{[1]*}\right)_{\alpha'_1} \left(\sum_{j=1}^{d} \left(W_{ij}^{[1]}\right)_{\beta_1} \left(M_j^{[1]}\right)_{\alpha_1}\right). \tag{2.134}$$

Then, any subsequent 3-tensor is computed with the following optimal contractions:

$$\begin{aligned}\left(L^{[s]}\right)_{\alpha'_s\beta_s\alpha_s} &= \sum_{i=1}^{d} \sum_{\alpha'_{s-1}=1}^{\widetilde{D}_{s-1}} \left(\widetilde{M}_i^{[s]*}\right)_{\alpha'_{s-1}\alpha'_s} \\ &\times \left(\sum_{j=1}^{d} \sum_{\beta_{s-1}=1}^{D_W} \left(W_{ij}^{[s]}\right)_{\beta_{s-1}\beta} \left(\sum_{\alpha_{s-1}=1}^{D_{s-1}} \left(L^{[s-1]}\right)_{\alpha'_{s-1}\beta_{s-1}\alpha_{s-1}} \left(M_j^{[s]}\right)_{\alpha_{s-1}\alpha_s}\right)\right).\end{aligned} \tag{2.135}$$

The cost of this operation is:

$$\mathcal{O}(dD_{s-1}D_s\widetilde{D}_{s-1}D_W) + \mathcal{O}(d^2D_s\widetilde{D}_{s-1}D_W^2) + \mathcal{O}(dD_s\widetilde{D}_{s-1}\widetilde{D}_sD_W). \tag{2.136}$$

This operation defines the transfer operator for matrix elements and expectation values of MPO:

$$\mathbb{E}_{\text{mpo}}^{[s]} = \sum_{i,j=1}^{d} \widetilde{M}_i^{[s]*} \otimes W_{ij}^{[s]} \otimes M_j^{[s]}. \tag{2.137}$$

By very similar considerations, it is easy to derive the rules to grow the right blocks, from $R^{[s]}$ to $R^{[s-1]}$, for matrix elements of MPOs, but we will not report them here.

2.13 Hamiltonians as MPOs

In this work, we decided to express every operator as an MPO. The most important operator is the hamiltonian of the problem, this means that we must be able to translate a human readable hamiltonian into a matrix product operator. Take a look at a many-body hamiltonian, such as the Ising chain with external field:

$$H = J\sum_{s=1}^{N-1} S_s^x S_{s+1}^x + h\sum_{s=1}^{N} S_s^z. \tag{2.138}$$

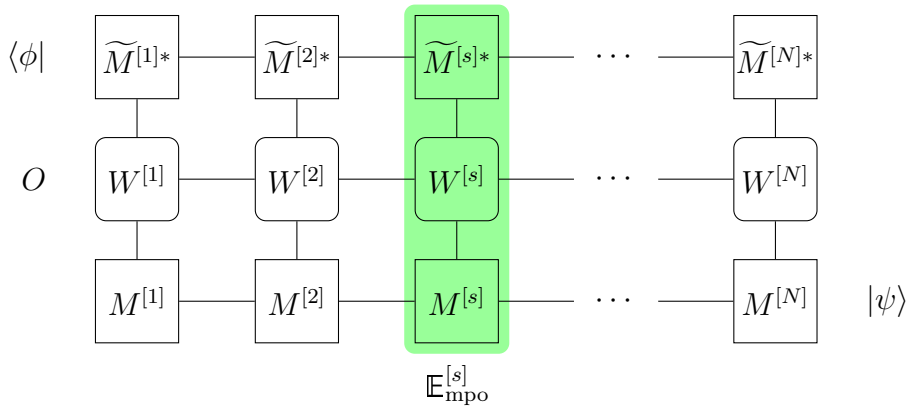


Figure 2.13: Tensor network for the matrix element $\langle\phi|O|\psi\rangle$, with highlighted transfer operator for MPOs.

At first sight, it seems impossible to write it as an MPO, but, in fact, the conversion, although not exactly trivial, is simple and elegant [55, 56].

Initially, we consider only the external field and ignore the coupling constant (for the moment). The term $\sum_s S^z_s$ is short for the sum of many strings of operators in the form:

$$\cdots \otimes \mathbb{1} \otimes S^z_s \otimes \mathbb{1} \otimes \mathbb{1} \otimes \cdots \tag{2.139}$$

where there are identity matrices everywhere except for the s-th site in which S^z is positioned[4]. It is useful to regroup the elements of the matrices of the MPO not following the virtual index but the physical index:

$$\widehat{W}^{[s]}_{\beta\beta'} = \sum_{i,j=1}^{d} (W^{[s]}_{ij})_{\beta\beta'} \, |i\rangle\langle j| \, . \tag{2.140}$$

Then, the MPO is a compact multiplication of operator-valued matrices of dimensions $D_W \times D_W$ (except the first and the last sites):

$$O = \widehat{W}^{[1]}\widehat{W}^{[2]} \cdots \widehat{W}^{[N]}. \tag{2.141}$$

Each $\widehat{W}^{[s]}$ acts on a different site of the chain, therefore the full multiplication results in a sum of tensor products of operators, exactly what we need.

When we want to write a string like (2.139) from right to left (for later convenience) we have to follow a rule. The rule is: write either $\mathbb{1}$ or S^z, but,

[4]For a small connection with computer science: strings of operators like these are equivalent to the strings of character that can be written by a Turing machine (using a finite number of states).

if the choice is S^z then complete the string with only $\mathbb{1}$ (the length of the string is fixed by N). We are able to identify two *states*: in the first one, if we write $\mathbb{1}$ we remain in state 1, if we write S^z we pass to state 2. In state 2 we are allowed to write only $\mathbb{1}$ and, then, to remain in state 2.

We encode state 1 with vector $\left(\begin{smallmatrix}1\\0\end{smallmatrix}\right)$ and state 2 with vector $\left(\begin{smallmatrix}0\\1\end{smallmatrix}\right)$, hence we find the following transformation matrix:

$$W = \begin{pmatrix} \mathbb{1} & 0 \\ S^z & \mathbb{1} \end{pmatrix}. \tag{2.142}$$

Given the vector space nature of the encoding that we chose, a few matrices like (2.142) multiplied together generate every possible string of the form (2.139), that is just what we desire. More precisely, we always begin with state 1 and end with state 2, this means that we have to enclose the multiplied matrices between $\left(\begin{smallmatrix}0&1\end{smallmatrix}\right)$ and $\left(\begin{smallmatrix}1\\0\end{smallmatrix}\right)$. It is easy to verify that with N multiplied matrices, we produce all the operator strings of kind (2.139) of length N:

$$\begin{aligned}\begin{pmatrix}0 & 1\end{pmatrix}\begin{pmatrix} \mathbb{1} & 0 \\ S^z & \mathbb{1} \end{pmatrix}\begin{pmatrix} \mathbb{1} & 0 \\ S^z & \mathbb{1} \end{pmatrix}\begin{pmatrix} \mathbb{1} & 0 \\ S^z & \mathbb{1} \end{pmatrix}\begin{pmatrix} \mathbb{1} & 0 \\ S^z & \mathbb{1} \end{pmatrix}\begin{pmatrix}1\\0\end{pmatrix} = \\ = \mathbb{1}\otimes\mathbb{1}\otimes\mathbb{1}\otimes S^z + \mathbb{1}\otimes\mathbb{1}\otimes S^z\otimes\mathbb{1} + \\ + \mathbb{1}\otimes S^z\otimes\mathbb{1}\otimes\mathbb{1} + S^z\otimes\mathbb{1}\otimes\mathbb{1}\otimes\mathbb{1}\end{aligned} \tag{2.143}$$

To complete the creation of the MPO, we put $\widehat{W}^{[s]} = W$ for every site s. First and last sites are again a special case, that make sure the coefficients of the MPO are scalars. The first site is a row matrix containing the last row of W, while the last site is a column vector containing the first column of W; it is a direct consequence of their multiplication with $\left(\begin{smallmatrix}0&1\end{smallmatrix}\right)$ and $\left(\begin{smallmatrix}1\\0\end{smallmatrix}\right)$.

Now we pass to the nearest-neighbor interaction: $\sum_s S^x_s S^x_{s+1}$. This time in the initial state we are allowed to write $\mathbb{1}$ or S^x; with S^x we pass to an intermediate state where we can only write another S^x and then go to final state in which we remain, writing only $\mathbb{1}$. The number of states is three so the transformation matrix is 3×3:

$$W = \begin{pmatrix} \mathbb{1} & 0 & 0 \\ S^x & 0 & 0 \\ 0 & S^x & \mathbb{1} \end{pmatrix}. \tag{2.144}$$

Putting everything together, we find the total transformation matrix of the Ising chain. From the initial state we can write S^z to go directly to the final state or S^x to pass through the intermediate state:

$$W = \begin{pmatrix} \mathbb{1} & 0 & 0 \\ S^x & 0 & 0 \\ S^z & S^x & \mathbb{1} \end{pmatrix}. \tag{2.145}$$

The coupling constants we ignored until now, should be multiplied just before passing to the final state (unless some more exotic behaviour is sought):

$$W = \begin{pmatrix} \mathbb{1} & 0 & 0 \\ S^x & 0 & 0 \\ hS^z & JS^x & \mathbb{1} \end{pmatrix}. \tag{2.146}$$

For a more convoluted example, consider the anisotropic Heisenberg chain with external field:

$$H = \sum_{s=1}^{N-1} \left(J_1 S_s^+ S_{s+1}^- + J_1 S_s^- S_{s+1}^+ + J_2 S_s^z S_{s+1}^z \right) + h \sum_{s=1}^{N} S_s^z, \tag{2.147}$$

whose transformation matrix is:

$$W = \begin{pmatrix} \mathbb{1} & 0 & 0 & 0 & 0 \\ S^+ & 0 & 0 & 0 & 0 \\ S^- & 0 & 0 & 0 & 0 \\ S^z & 0 & 0 & 0 & 0 \\ hS^z & J_1 S^- & J_1 S^+ & J_2 S^z & \mathbb{1} \end{pmatrix}. \tag{2.148}$$

We call every MPO whose operator-valued matrices $\widehat{W}^{[s]}$ are the repetition of the same matrix W a (*site*) *invariant MPO*.

Chapter 3

Ground state algorithms

The problem of ground state search is to find the state $|\psi\rangle$ that minimizes the quantity:

$$E = \frac{\langle\psi|H|\psi\rangle}{\langle\psi|\psi\rangle}, \tag{3.1}$$

where H is the hamiltonian of the system. In this chapter, we present the new algorithm that we devised and tested in our work, but before we introduce the density matrix renormalization group, an important algorithm for ground state search that we will use for comparison with our method.

3.1 Density Matrix Renormalization Group

The Density Matrix Renormalization Group (or DMRG) is the algorithm for ground state search introduce by Steven White [57] (1992). It has become the most widely used method to find ground states of strongly correlated quantum many-body systems. In this section, we will briefly introduce it, explaining how it works and how it is implemented in MPS language. In chapter 4 we will compare the new algorithm presented in this work at section 3.2 with this established procedure. Here, we present DMRG as it is computed as a variational research on matrix product states, not in its original formulation.

As we know from chapter 1 and chapter 2, for gapped, local hamiltonians, the ground state can be found in a much narrower class of states: the MPS class (confront with equations (2.1) to (2.3)). Now, to search the ground state, we add a langrangian multiplier λ and we have to extremize:

$$\langle\psi|H|\psi\rangle - \lambda\,\langle\psi|\psi\rangle\,, \tag{3.2}$$

with $|\psi\rangle$ in MPS form and H in MPO form (see section 2.10). The parameters are all the elements of the matrices of $|\psi\rangle$ (plus λ). Being all these

variables multiplied together, this is a highly non-linear problem. To reduce its complexity, it is possible to fix the value of all the parameters but those of the matrices of a single site s. Then, the elements of the matrices of site s appear in (3.2) only quadratically and, when we extremize the expression, the problem becomes linear. Once we have found the values that minimize the energy E, we move to another site t, fix all the parameters not on site t and minimize the energy again. This is an iterative procedure that always finds a lower or equal E and that proceeds variationally towards the ground state.

Using the expressions of sections 2.7.1 and 2.12, we will write equation (3.2) in a more compact form, leaving the matrices of site s explicit. We call the left and right blocks of the overlap as $\widetilde{L}^{[s-1]}$ and $\widetilde{R}^{[s+1]}$, while left and right blocks of the MPO expectation value are $L^{[s-1]}$ and $R^{[s+1]}$. Then, with the help of the graphical representation in figure 3.1, we have for the overlap:

$$\begin{aligned}\langle\psi|\psi\rangle = \sum_{i_s=1}^{d} \sum_{(\alpha_{s-1},\alpha'_{s-1}=1)}^{D_{s-1}} \sum_{\alpha_s,\alpha'_s=1}^{D_s} (\widetilde{L}^{[s-1]})_{\alpha'_{s-1}\alpha_{s-1}} \\ \times (M^{[s]}_{i_s})_{\alpha_{s-1}\alpha_s} (\widetilde{R}^{[s+1]})_{\alpha_s\alpha'_s} (M^{[s]*}_{i_s})_{\alpha'_{s-1}\alpha'_s} = \\ = \sum_{i_s=1}^{d} \mathrm{Tr}\Big[M^{[s]\dagger}_{i_s} \widetilde{L}^{[s-1]} M^{[s]}_{i_s} \widetilde{R}^{[s+1]} \Big]; \quad (3.3)\end{aligned}$$

while for the MPO expectation value:

$$\begin{aligned}\langle\psi|H|\psi\rangle = \sum_{i_s,j_s=1}^{d} \sum_{(\beta_{s-1},\beta_s=1)}^{D_H} \sum_{(\alpha_{s-1},\alpha'_{s-1}=1)}^{D_{s-1}} \sum_{\alpha_s,\alpha'_s=1}^{D_s} (L^{[s-1]})_{\alpha'_{s-1}\beta_{s-1}\alpha_{s-1}} \\ \times (M^{[s]}_{j_s})_{\alpha_{s-1}\alpha_s} (W^{[s]}_{i_s j_s})_{\beta_{s-1}\beta_s} (R^{[s+1]})_{\alpha_s\beta_s\alpha'_s} (M^{[s]*}_{i_s})_{\alpha'_{s-1}\alpha'_s}. \quad (3.4)\end{aligned}$$

To extremize (3.2) to find the elements $(M^{[s]}_{i_s})_{\alpha_{s-1}\alpha_s}$ that minimize the energy, we have to derivate with respect to $(M^{[s]*}_{i_s})_{\alpha'_{s-1}\alpha'_s}$. After the derivation, the corresponding M^* pieces disappear from the equations and the graphs.

At this point we decide to take $|\psi\rangle$ in mixed-canonical form such that all the matrices to the left of site s are left-normalized and all the matrices to the right of site s are right-normalized. In turns, this implies that blocks $\widetilde{L}^{[s-1]}$ and $\widetilde{R}^{[s+1]}$ reduce to identity matrices. Of the derivative of equation (3.2), it remains:

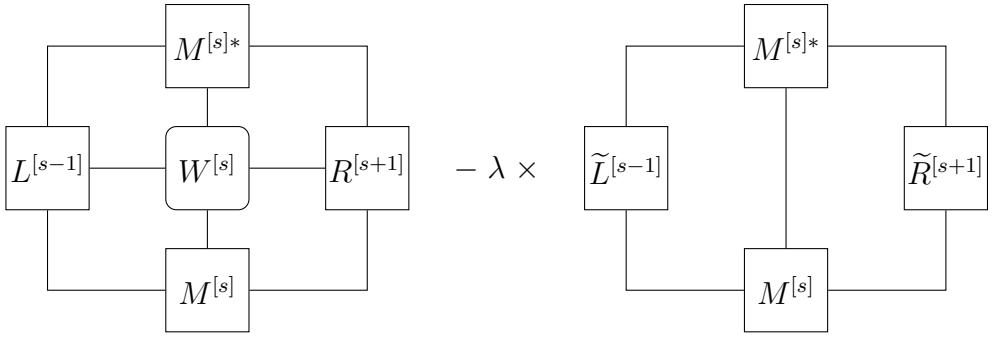


Figure 3.1: The full network of contractions of $\langle\psi|H|\psi\rangle - \lambda\langle\psi|\psi\rangle$.

$$\sum_{j_s=1}^{d}\sum_{(\beta_{s-1},\beta_s=1)}^{D_H}\sum_{(\alpha_{s-1}=1)}^{D_{s-1}}\sum_{\alpha_s=1}^{D_s}(L^{[s-1]})_{\alpha'_{s-1}\beta_{s-1}\alpha_{s-1}} \times (W^{[s]}_{i_s j_s})_{\beta_{s-1}\beta_s}(R^{[s+1]})_{\alpha_s\beta_s\alpha'_s}(M^{[s]}_{j_s})_{\alpha_{s-1}\alpha_s} - \lambda(M^{[s]}_{i_s})_{\alpha'_{s-1}\alpha'_s}. \quad (3.5)$$

We regroup the tensors LWR as a single big matrix that we call $\mathbb{H}$ by analogy to the hamiltonian:

$$(\mathbb{H})_{(i_s\alpha'_{s-1}\alpha'_s),(j_s\alpha_{s-1}\alpha_s)} = = \sum_{(\beta_{s-1},\beta_s=1)}^{D_H}(L^{[s-1]})_{\alpha'_{s-1}\beta_{s-1}\alpha_{s-1}}(W^{[s]}_{i_s j_s})_{\beta_{s-1}\beta_s}(R^{[s+1]})_{\alpha_s\beta_s\alpha'_s}, \quad (3.6)$$

and we unwrap the matrix elements $(M^{[s]}_{i_s})_{\alpha'_{s-1}\alpha'_s}$ as a single long vector $\mathbf{v}$:

$$(\mathbf{v})_{(i_s\alpha'_{s-1}\alpha'_s)} = (M^{[s]}_{i_s})_{\alpha'_{s-1}\alpha'_s}; \quad (3.7)$$

$\mathbf{v}$ has length $dD_{s-1}D_s$, while $\mathbb{H}$ has dimensions $dD_{s-1}D_s \times dD_{s-1}D_s$. In this way, we have translated the minimization of (3.2) to the solution of the eigenvalue problem of $\mathbb{H}$:

$$\mathbb{H}\mathbf{v} - \lambda\mathbf{v} = 0 \quad (3.8)$$

Since $\mathbb{H}$ is very large and we are only interested in the lowest energy eigenvector, the problem (3.8) is treated with an iterative eigensolver like the Lanczos algorithm or the Jacobi-Davidson algorithm.

3.2 Kicker strategy

In this section we present the new contribution of this work: the proposal of a new algorithm for ground state search, that we call *kicker strategy*. The

starting point is radically different: DMRG exploits the MPS structure to improve locally, matrix by matrix, the guess of the ground state. Our point of view is more "vectorial" and we globally change the whole state at every step. Matrix product states are just used as the natural language to encode enormous many-body vector states. The idea is inspired by the power method and a very simplified version of the Lanczos method [16].

3.2.1 Outline

Initially we start from a state $|\psi_0\rangle$, normalized to one and chosen randomly, if we have no other clue. Then we apply an operator K, called the "*kicker*", to produce the state $K|\psi_0\rangle$ and we orthonormalize it with respect to $|\psi_0\rangle$:

$$|\psi_\perp\rangle \coloneqq \frac{K - \langle K\rangle_0}{\sqrt{\langle K^\dagger K\rangle_0 - |\langle K\rangle_0|^2}} |\psi_0\rangle \,. \tag{3.9}$$

We used the notation $\langle A\rangle_0 \coloneqq \langle\psi_0|A|\psi_0\rangle$. The two orthonormal vectors, $|\psi_0\rangle$ and $|\psi_\perp\rangle$, span a two-dimensional subspace where we diagonalize the *reduced hamiltonian*:

$$\begin{pmatrix} \langle\psi_0|H|\psi_0\rangle & \langle\psi_0|H|\psi_\perp\rangle \\ \langle\psi_\perp|H|\psi_0\rangle & \langle\psi_\perp|H|\psi_\perp\rangle \end{pmatrix} =: \begin{pmatrix} E_0 & \gamma \\ \gamma^* & E_\perp \end{pmatrix}. \tag{3.10}$$

Say that the column vector $\binom{\alpha}{\beta}$ is the lowest energy eigenvector of the reduced hamiltonian, then we define:

$$|\psi_1\rangle = \alpha|\psi_0\rangle + \beta|\psi_\perp\rangle \,, \tag{3.11}$$

and, as it is easily verified, $E_1 \coloneqq \langle\psi_1|H|\psi_1\rangle \leq E_0$, therefore $|\psi_1\rangle$ is a better approximation of the ground state. We can iterate the previous steps to produce a sequence of states: $\{|\psi_k\rangle\}_k$, with decreasing energies: $E_{k+1} \leq E_k, \forall k$. In this sense the procedure is variational, we never go up in energy, but we have no guarantee that we can actually reach the ground state.

It is clear that the efficiency of this procedure rely heavily on a wise choice of the kicker K. In particular, we wish the kicker to have the following naive properties:

- the ground state is not excluded from its image;
- it makes the process work with every possible initial state;
- it does not get stuck, meaning that it doesn't favour a subspace orthogonal to the ground state;

- it is a bounded operator, otherwise the new state could have infinite norm.

Pointing towards the ground state is not a necessary prerequisite for the kicker since we just want to know a "new" direction where to look for, then we hope that the diagonalization of the reduced hamiltonian is helpful enough to descend towards the ground state. Part of this work will be devoted to compare different families of kickers and their efficiency.

All the information we want to extract is contained in the hamiltonian H that is also the only object we are dealing with, so it is expected that a good choice of kicker will be a function of the hamiltonian: $K = K(H)$. On the other hand, at this point we cannot neglect the possibility that a random operator will be more efficient.

3.2.2 Technical details

The description of the algorithm we gave above is abstract and not suited for a concrete implementation on a machine. Indeed we never even mentioned how the various objects, such as $|\psi_0\rangle$ or K, are saved in memory. To answer this question right away: the guesses of the ground state, namely $|\psi_0\rangle$, $|\psi_1\rangle$ and all the following approximations, are saved as Matrix Product States because of the reasons we already discussed about the large number of degrees of freedom and the presence of area laws. To maintain the MPS structure, and also for many other interesting features, we decided to save the hamiltonian H and the kicker K as Matrix Product Operators. Among these features we can describe longer-than-nearest-neighbor interactions and also quick and efficient building of new operators, through products, sums and, most likely, compression.

Since the vectors are very large even in MPS form, it is very good to have the minimum number of them in memory. The minimum but not the least, because saving the result of very demanding computations may speed up the algorithm avoiding continuous recalculations. For our purpose, for example, it is necessary to keep in memory $|\psi_0\rangle$ and $K|\psi_0\rangle$; the latter requires a heavy computation indeed. On the other hand, it is inefficient to compute $|\psi_\perp\rangle$, because the orthonormalization implies a sum and sums are costly for consecutive operations, as they increase the bond link dimension. To circumvent the problem we express everything in terms of only $|\psi_0\rangle$ and $K|\psi_0\rangle$.

Before listing the steps of the algorithm, we discuss in details the construction of various entities, beginning with $K|\psi_0\rangle$: obviously, it requires an MPO·MPS product, this heavily increases the bond link. To avoid that the

bond link of the vector state reaches a prohibitive value, we limit at crucial steps the maximum value it can take. Say that, before being multiplied by the kicker, $|\psi_0\rangle$ has maximum bond link dimension $D^{\max}$, and that K has bond link dimension D_K. Then, $K|\psi_0\rangle$ has bond dimension $D_K D^{\max}$, with D_K in the order of ten. To bring back the bond link to $D^{\max}$ the optimal method is the variational compression (see section 2.9.2) suited for large order of compression.

Substituting the definition of $|\psi_\perp\rangle$ of equation (3.9), in (3.10), we find how E_0, γ and $E_\perp$ are defined in terms of only $|\psi_0\rangle$ and $K|\psi_0\rangle$:

$$\begin{aligned} E_0 &= \langle H\rangle_0 , \\ \gamma &= \frac{\langle HK\rangle_0 - \langle H\rangle_0 \langle K\rangle_0}{\sqrt{\langle K^\dagger K\rangle_0 - |\langle K\rangle_0|^2}}, \\ E_\perp &= \frac{\langle K^\dagger HK\rangle_0 + \langle H\rangle_0 |\langle K\rangle_0|^2 - 2\Re(\langle HK\rangle_0 \langle K\rangle_0^*)}{\langle K^\dagger K\rangle_0 - |\langle K\rangle_0|^2}. \end{aligned} \tag{3.12}$$

Thus the knowledge of $\langle H\rangle_0$, $\langle K\rangle_0$, $\langle HK\rangle_0$, $\langle K^\dagger K\rangle_0$ and $\langle K^\dagger HK\rangle_0$, lets us calculate the three matrix elements. The two eigenvalues of (3.10) are:

$$\lambda_\pm = \frac{E_0 + E_\perp}{2} \pm \frac{1}{2}\sqrt{(E_0 - E_\perp)^2 + 4|\gamma|^2}; \tag{3.13}$$

and the lowest one is $E_1 = \lambda_-$, then very easily we have $E_1 \leq E_0$ whatever the value of $E_\perp$ and γ. The eigenvector $\binom{\alpha}{\beta}$ corresponding to the lowest eigenvalue has the following two components:

$$\begin{aligned} \alpha &= \frac{\gamma}{\sqrt{|\gamma|^2 + (E_1 - E_0)^2}}, \\ \beta &= \frac{E_1 - E_0}{\sqrt{|\gamma|^2 + (E_1 - E_0)^2}}. \end{aligned} \tag{3.14}$$

At this point we can write:

$$|\psi_1\rangle = a|\psi_0\rangle + bK|\psi_0\rangle , \tag{3.15}$$

where:

$$\begin{aligned} a &= \frac{\alpha - \beta\langle K\rangle_0}{\sqrt{\langle K^\dagger K\rangle_0 - |\langle K\rangle_0|^2}}, \\ b &= \frac{\beta}{\sqrt{\langle K^\dagger K\rangle_0 - |\langle K\rangle_0|^2}}. \end{aligned} \tag{3.16}$$

This is an MPS+MPS sum and, again, the bond link grows, but only twice the original size (since both $|\psi_0\rangle$ and $K|\psi_0\rangle$ have bond link dimension of $D^{\max}$ at most). The appropriate shrinking method is now the SVD compression (see section 2.9.1).

After all these considerations we are ready to list the steps of a more functional algorithm:

Algorithm 3.1: Kicker strategy

1. create the hamiltonian MPO H as described in section 2.13;
2. build the kicker MPO K (the details of this part will be elaborated in the next sections);
3. generate a random MPS $|\psi_0\rangle$ and normalize it to 1;
4. apply the kicker to the vector state to obtain $K|\psi_0\rangle$;
5. compress variationally $K|\psi_0\rangle$;
6. compute the expectations: $\langle H\rangle_0$, $\langle K\rangle_0$, $\langle HK\rangle_0$, $\langle K^\dagger K\rangle_0$ and $\langle K^\dagger HK\rangle_0$;
7. compute the elements of the reduced hamiltonian, E_0, γ and $E_\perp$ as in (3.12) and from them the lowest energy eigenvalue E_1, the coefficients α, β and a, b (respectively equations (3.13), (3.14) and (3.16));
8. sum the two saved MPS to produce $|\psi_1\rangle$ as in (3.15);
9. compress $|\psi_1\rangle$ with SVD;
10. return to point 4 with $|\psi_0\rangle \leftarrow |\psi_1\rangle$.

As for any useful calculation, there must be a stop. There are mainly three stopping mechanisms:

- fixing the maximum number of iterations: this is very rigid and offers no versatility; anyway it is very useful as a global override and it avoids the program spending too much time stuck on a problem that may have just wrong parameters;
- checking the energy value $E_1 = \langle H\rangle_1$: this is more desirable since it dynamically stops when a fixed point is reached; the energy is already calculated every step so this test has no additional costs; a possibility

is to compare the current energy with the previous one and, if the difference is smaller than ϵ for more than t times, then the algorithm terminates;

- checking the energy variance $\langle H^2 \rangle_1 - \langle H \rangle_1^2$: this is a finer test for convergence as it reveals if an eigenstate of the hamiltonian is reached; the value must be as close to zero as possible, so a similar strategy as that proposed in the previous point with the energy difference works fine; a downside is the greater cost due to the square of the hamiltonian, but the cost could be lowered saving in memory the MPO of the square.

3.2.3 Analysis

For future comparison with other similar algorithms, in this section we account for every part's complexity.

We do not include in the count the creation of the hamiltonian, of the kicker and of the random initial vector (points 1, 2 and 3 from algorithm 3.1), because they happen only once. Point 4 is an MPO·MPS product that costs $\mathcal{O}(Nd^2 D_K^2 D^2)$. Then comes the variational compression of $K|\psi\rangle$ (point 5) whose complexity we already addressed at the end of section 2.9.2. Substitute the value of $\widetilde{D} = D_K D$ in (2.125):

$$\mathcal{O}(\chi_v N d D (D + D_K D)^2) = \mathcal{O}(\chi_v N d D^3 D_K^2). \tag{3.17}$$

The expectation values of point 6 deserve some discussion. One might be tempted to save the MPS $H|\psi_0\rangle$ and to compute $\langle H \rangle_0$ and $\langle HK \rangle_0$ as overlaps to spare some time. We call this course method 1, while the choice to compute them as matrix elements of H is named method 2. Unfortunately $H|\psi_0\rangle$ cannot be compressed, because otherwise the orthonormality of $|\psi_\perp\rangle$ would be compromised and the reduced hamiltonian would not have the correct values. We tested both methods against a limited number of cases starting from the same random vector and we could not find any significant difference. Anyway, a careful analysis of the complexity shows that method 2 is more efficient asymptotically. From equations (2.90), (2.131) and (2.136) we apprehend that method 1 has cost:

$$\mathcal{O}(Nd^2 D^2 D_H^2) + \mathcal{O}(NdD^3 D_H^2), \tag{3.18}$$

while method 2 has cost:

$$\mathcal{O}(Nd^2 D^2 D_H^2) + \mathcal{O}(NdD^3 D_H). \tag{3.19}$$

The explanation is simple: the matrix elements of an MPO (see section 2.12) are already optimally contracted. Trying to evaluate the same tensor network

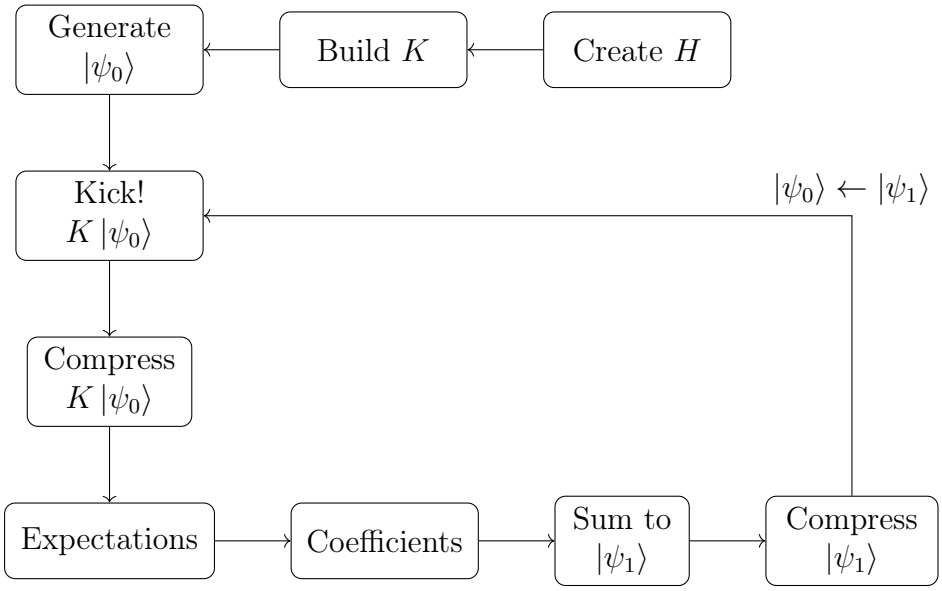


Figure 3.2: The flow diagram of the Kicker stategy algorithm.

with a different method, such as saving the MPO·MPS product and then compute the overlap, will have an equal or greater cost. Needless to say, we chose method 2.

Point 7 is $\mathcal{O}(1)$ and the sum of point 8 takes a negligible amount of operations being just a copy. Finally, the SVD compression in point 9 has cost of $\mathcal{O}(NdD^3)$, because the bond link dimension of the sum is $2D$ and it needs a complete canonization. The important result we obtained is that every step of the algorithm requires:

$$\mathcal{O}(\chi_v N d D^3 D_K^2) + \mathcal{O}(N d^2 D^2 D_H^2) + \mathcal{O}(N d D^3 D_H). \tag{3.20}$$

Probably the last term could be dropped because, usually, $D_K \geq D_H$.

3.3 Hamiltonian kicker

The hamiltonian kicker (HK) is the simplest kicker we can devise: we simply put $K = H$. This kicker has several advantages:

- it has bond link dimensions equal to those of the hamiltonian and we will see that this means they are very small compared to other kickers;
- a smaller bond link means a faster multiplication and compression (for the kick);

- it also means that for very large hamiltonians (such as those with longer-than-nearest-neighbor interactions) or hamiltonians whose powers are hard to compress, this is the only manageable kicker we have.

One of the reasons to use this kicker is that famous iterative eigensolvers, like the power method or the Lanczos algorithm [16] and related procedures, use the same matrix they want to find the eigenvalues of, to repeatedly multiply the same vector.

We already know that $E_1 \leq E_0$ for every possible kicker, but here we can give a few more considerations. In the hamiltonian kicker case, both the norm of $(K - \langle K \rangle) |\psi_0\rangle$ (to obtain $\psi_\perp$) and γ are equal to ΔE_0, i.e. the square root of the variance of energy of $|\psi_0\rangle$: $\Delta E_0 = \sqrt{\langle H^2 \rangle_0 - \langle H \rangle_0^2}$. This has two undesirable consequences:

- we cannot start from (or pass through) any eigenvector of H, otherwise we are not able to compute $|\psi_\perp\rangle$;
- the distance between E_0 and E_1 is a monotone increasing function of $|\gamma| = \Delta E_0$: the nearer we are to an eigenvalue, the smaller the value of the variance will be; consequently, we are doomed to converge only asymptotically to the ground state.

Unfortunately this is valid for *every* kicker that is a function of the hamiltonian. This clashes with the fact that a function of the hamiltonian is also the best shot we have to quickly find the ground state, as explained in the next section. A possible solution to this problem could be to switch to another kicker once we are near to an eigenvalue; of course the other kicker must have eigenvalues different from those of H.

3.4 Approximate ground state projections

The hamiltonian kicker is very good, but it is only our first attempt, maybe there is an even better kicker. Indeed, although we said that pointing towards the ground state is not a fundamental requisite, it surely is of great help. The strategy is to think in terms of spectrum.

We know that every state is expandable on a complete orthonormal basis formed by the eigenvectors $|w_k\rangle$ of H, so even $|\psi_0\rangle$:

$$|\psi_0\rangle = \sum_{k=0}^{d^N-1} c_k |w_k\rangle , \qquad H |w_i\rangle = \varepsilon_i |w_i\rangle , \qquad i < j \implies \varepsilon_i < \varepsilon_j . \quad (3.21)$$

Some notation choices: the ground state energy is ε_0, the gap with the first excited state is $g = \varepsilon_1 - \varepsilon_0$ and we indicate the maximum eigenvalue as $u = \varepsilon_{d^N-1}$, so that the spectrum is included in the interval $[\varepsilon_0, u]$.

The spectral theorem says that any function $f(x)$, that is approximable by polynomials, defines an operator $f(H)$ whose eigenvectors are at least those of H and whose eigenvalues are the image of the original eigenvalues:

$$f(H)\,|w_i\rangle = f(\varepsilon_i)\,|w_i\rangle\,. \tag{3.22}$$

The ideal function to build is the characteristic function of an interval I we suppose contains the ground state ($\varepsilon_0 \in I$):

$$\chi_I(x) = \begin{cases} 1 & \text{if } x \in I \\ 0 & \text{otherwise} \end{cases}. \tag{3.23}$$

If the length of the interval is small enough to exclude every other eigenvalue ($\varepsilon_i \notin I, \forall i > 0$), we can actually build directly the projection on the ground state: $\chi_I(H) = \Pi_0$, $\Pi_0\,|w_0\rangle = |w_0\rangle$, $\Pi_0\,|w_i\rangle = 0$, for $i \neq 0$. Alternatively, if we do not know the exact value of ε_0 (and g), we could try to build the inverse Heaviside function:

$$\theta(-x) = \begin{cases} 1 & \text{if } x \leq 0 \\ 0 & \text{otherwise} \end{cases}, \tag{3.24}$$

shifting it after every step a bit more towards the ground state energy: $\theta_a(x) = \theta(-(x-a)), a \to \varepsilon_0$.

Unfortunately, this is not possible, since these functions possess discontinuities and plateaux that are hard to approximate. Indeed, just to get near them, it would take a polynomial of the hamiltonian with a prohibitive degree in MPO language. Nevertheless, we could try to mimic something simpler but with similar behaviour. This is why we introduce the following definition[1]:

Definition 3.1. *We say that an operator K is a Δ-Approximate Ground Space Projection (AGSP) for H if it is true that:*

- *any ground state $|w_0\rangle$ is invariant under this operator: $K\,|w_0\rangle = |w_0\rangle$;*
- *any state $|\phi\rangle$ orthogonal to ground state, $\langle w_0|\phi\rangle = 0$, remains orthogonal, $\langle w_0|K|\phi\rangle = 0$, and its norm is reduced by a factor Δ: $\|K\,|\phi\rangle\|^2 \leq \Delta$.*

If K is a function of the hamiltonian, we do not have to worry about the orthogonality with the ground state (second point). In the next two section we will see two examples of AGSP that are function of the hamiltonian.

[1] directly from [17]

3.5 Imaginary time evolution kicker

The imaginary time evolution kicker (ITEK) is not a single operator but a class of kickers, spanned by two parameters, one continuous and the other discrete.

As we know, time evolution in non-relativistic quantum mechanical systems is described by the application of the operator e^{-itH} to the state vector. If we choose $t = i\beta$, with β real, then the operator becomes $e^{-\beta H}$. Look at the action of this operator on the eigenvector expansion of $|\psi_0\rangle$:

$$e^{-\beta H}|\psi_0\rangle = \sum_{k=0}^{d^N-1} c_k e^{-\beta\varepsilon_k}|w_k\rangle \xrightarrow{\beta\to\infty} c_0 e^{-\beta\varepsilon_0}|w_0\rangle\,. \tag{3.25}$$

Again, if we could use an infinite β, we would be able to construct a ground state projector (after the normalization of the result), but an infinite β implies an infinitely accurate approximation of $e^{-\beta H}$ and we do not have the resources for that. Anyway, even with a small β, the expectation $\langle\psi_0|e^{-\beta H}He^{-\beta H}|\psi_0\rangle$ is lower that E_0, because the coefficients c_k, of equation (3.21), are weighted with a decreasing exponential function and their expectation value is brought down towards the ground state. This is not exactly the same definition as AGSP, but it works in a very similar fashion.

Our goal is to construct an approximation of the operator $e^{-\beta H}$ for the relevant values of β that could benefit us. Fortunately, in literature there are many methods designed for this task [58, 13, 59, 14]. This is one of the reasons why we immediately tried to build this kicker. Usually, these methods use the Trotter decomposition of an exponential operator, that at first order is:

$$e^{\delta(A+B)} \approx e^{\delta A}e^{\delta B} + \mathcal{O}(\delta^2), \qquad \delta \ll 1. \tag{3.26}$$

Very roughly, the idea is: if β is sufficiently small, at first order we could act with each term of the hamiltonian (at the exponent) on the sites where that term has support. If the hamiltonian has only nearest-neighbor interactions, then there are easy extensions to higher orders, that often are needed for more accuracy. Because of the limitation to only nearest-neighbor interactions, we chose another path for time evolution.

In [60], we found a short review of the methods that expand time evolution to interactions with longer range. We decided to implement the truncated Taylor expansion of the exponential with MPOs. Here the maximum power of the truncated Taylor expansion is the discrete parameter for the class of ITEK (the other parameter is β). Look at the expansion:

$$e^{-\beta x}|_p = \sum_{k=0}^{p} \frac{(-\beta x)^k}{k!} = 1 - \beta x + \frac{(\beta x)^2}{2} + \cdots + \frac{(\beta x)^p}{p!}; \tag{3.27}$$

this will be a reliable approximation of the exponential only for a bounded interval of the real variable x. The smaller the β, the wider the interval, of course, but to be more quantitative, we try to bound the error in the approximation by ϵ:

$$\left|e^{-\beta x} - e^{-\beta x}|_p\right| \leq \epsilon. \tag{3.28}$$

If ϵ is small enough, we can safely assume that the greatest contribution to the difference will come from the term with the lowest power in the rest of the series:

$$\left|e^{-\beta x} - e^{-\beta x}|_p\right| \approx \frac{|\beta x|^{p+1}}{(p+1)!} \leq \epsilon. \tag{3.29}$$

Inverting the inequality, we obtain the range of validity of the approximation with β fixed and within error ϵ:

$$|x| \leq \frac{\sqrt[p+1]{\epsilon(p+1)!}}{|\beta|} =: r_{\exp}. \tag{3.30}$$

Therefore, the truncated operator $e^{-\beta H}|_p$ will be a polynomial of degree p in H, that will act like a decreasing exponential only on those eigenvalues of H that are within range (3.30) (see also figure 3.3). If we have any clue about the spectrum, it is better to use it to make sure that all the eigenvalues are within the range of validity: $[\varepsilon_0, u] \subset [-r_{\exp}, r_{\exp}]$. Otherwise, we just have to use a sufficiently small β.

This need of information about the spectrum is a constant for more refined kickers. It is a disadvantage compared to the hamiltonian kicker, that does not need this kind of knowledge of the system, but we can always majorize the spectrum with the norm of the hamiltonian $\|H\|$. The norm could give very overestimated values for frustrated systems, but we think it is a good starting point for the simpler systems.

The algorithm to construct the MPO corresponding to the imaginary time evolution kicker is found in [60]:

Algorithm 3.2: Imaginary time evolution kicker, MPO construction

1. accept a real number β, a positive integer $p \geq 0$ and an MPO of the hamiltonian H;
2. create the MPO for the identity operator I (it has bond link dimension $D_I = 1$);
3. copy I to the MPO E that will become the truncated exponential

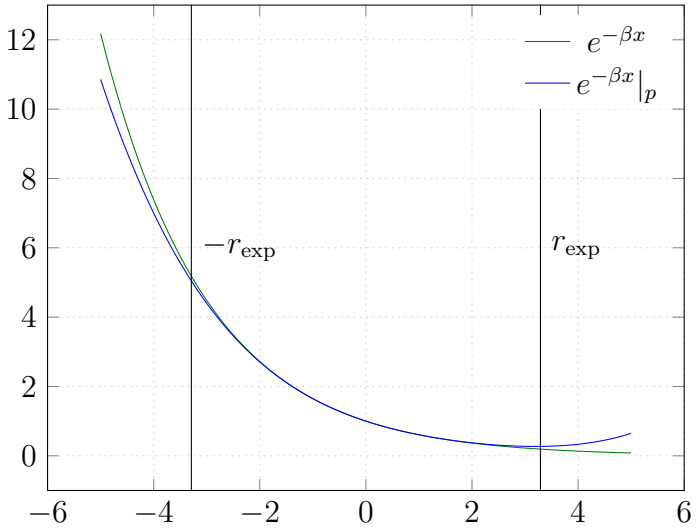


Figure 3.3: The decreasing exponential with $\beta = 1/2$ and its truncation to degree $p = 4$. The range of validity of the approximation ($[-r_{\exp}, r_{\exp}]$) is shown with $\epsilon = 0.1$ (a 10% error).

in the end: $E = I$;

4. while $p > 0$, do the following steps:
   (a) compute the MPO·MPO product $T = E \cdot H$;
   (b) compute the MPS+MPS sum $E = I + \frac{T}{p}$;
   (c) perform an SVD compression on E;
   (d) return to point 4 with $p \leftarrow p - 1$.

We see that there are a lot of operations that increase the bond link of E. Indeed, for a hamiltonian MPO of bond link dimension $D_H = 5$ and power $p = 3$ we would already have an MPO for E of bond link dimension $D_E = 156$. This is the reason why we put an SVD compression at point 4c. Usually, the powers of H do not need all the theoretically maximum bond link and the size of E is again manageable for a kick.

3.6 Chebyshev kicker

The Chebyshev kicker (CK) is another class of operators, based on Chebyshev polynomials. The reason to study this kinds of kickers is that Chebyshev polynomials are optimal in a sense that we will try to make clear later. For the moment, let us just say that, if we are going to use a polynomial of the hamiltonian as a kicker, among all the polynomials of degree p (fixed), it is worth considering to use a Chebyshev polynomial, because they better suppress the coefficients of higher eigenvalues' eigenvectors in our current state. The proof that Chebyshev polynomials define very good AGSPs comes from a lemma in [17]; our idea to adapt and implement the construction of such AGSPs in MPO language, for our ground state search algorithm, arose from there.

3.6.1 Chebyshev polynomials

Chebyshev polynomials are very important in many areas of mathematics and applied mathematics. They have very nice properties and they are particularly suitable for polynomial approximation. Information about them and all the notions presented below may be found in [61].

Definition 3.2 (Chebyshev polynomial)**.** *The Chebyshev polynomial of the first kind of degree p, is the only polynomial that satisfies:*

$$T_p(\cos\theta) = \cos(p\theta). \tag{3.31}$$

We find by direct calculation the Chebyshev polynomials for the first few degrees:

$$T_0(x) = 1 \tag{3.32}$$
$$T_1(x) = x \tag{3.33}$$
$$T_2(x) = 2x^2 - 1 \tag{3.34}$$
$$T_3(x) = 4x^3 - 3x \tag{3.35}$$

For higher degrees, it is easy to prove, by means of trigonometric formulae, the following important recursive relation:

$$T_p(x) = 2x\,T_{p-1}(x) - T_{p-2}(x). \tag{3.36}$$

We note from the definition (3.31) that:

$$|T_p(x)| \leq 1 \qquad \text{for } |x| \leq 1. \tag{3.37}$$

Of course, the Chebyshev polynomials are defined also for $|x| > 1$, where the substitution $x = \cos\theta$ is not valid anymore. Solving this last equation for complex θ, we find that there are other kinds of relations like (3.31) valid outside $|x| \leq 1$. For example, for the interval $x \in [1, +\infty)$ we have a similar identity for cosh that is satisfied by the same Chebyshev polynomial:

$$T_p(\cosh\Theta) = \cosh(p\Theta). \tag{3.38}$$

The property of Chebyshev polynomials that renders them optimal for our kicker strategy (and for many other uses) is the following: among all the polynomials $P_p(x)$ of degree p that satisfy $|P_p(x)| \leq 1$ for $|x| \leq 1$, $T_p(x)$ is the one with the highest leading coefficient (the coefficient of x^p). This coefficient is 2^{p-1} as deduced by (3.36). In other words, they are the polynomials that grow faster outside $|x| \leq 1$ being also bounded by 1 inside.

3.6.2 Chebyshev AGSP

From theorem 3.1, we understand that a kicker K, that is a function of the hamiltonian, $K = K(H)$, defines a proper AGSP if the function $K(x)$ has the following properties:

- $K(\varepsilon_0) = 1$;
- $|K(x)| \leq \sqrt{\Delta} < 1$ for $\varepsilon_1 \leq x \leq u$;

remembering that ε_0, ε_1 and u are, respectively, the ground state energy, the first excited state energy and the upper bound of the spectrum.

Indeed, it is possible to define such function with the use of Chebyshev polynomials [17]:

Theorem 3.3. *For every positive integer p, there exists a degree p polynomial C_p that satisfies the above conditions for:*

$$\sqrt{\Delta} = 2\exp\left(-2p\sqrt{(\varepsilon_1 - \varepsilon_0)/(u - \varepsilon_0)}\right). \tag{3.39}$$

Proof. The idea is to map the interval $[\varepsilon_1, u]$ to $[-1, 1]$, where we know there is a Chebyshev polynomial of degree p bounded by 1. Then we rescale the function such that it has value 1 for $x = \varepsilon_0$. Being ε_0 outside $[\varepsilon_1, u]$, the rescaling will compress the value of the function in $[\varepsilon_1, u]$, which is exactly what we desire. For a quantitative estimation of the compression Δ, consider $x > 1$ and map $x = \cosh\Theta$, with $\Theta \geq 0$. From equation (3.38), we obtain:

$$T_p(x) = \cosh(p\Theta) \geq \frac{1}{2}e^{p\Theta}. \tag{3.40}$$

Since we have the following inequality:

$$\Theta \geq 2\tanh\left(\frac{\Theta}{2}\right) = 2\sqrt{\frac{\cosh\Theta - 1}{\cosh\Theta + 1}} = 2\sqrt{\frac{x-1}{x+1}}, \tag{3.41}$$

we find:

$$T_p(x) \geq \frac{1}{2}e^{2p\sqrt{(x-1)/(x+1)}}. \tag{3.42}$$

Now consider function $f(x)$ like:

$$f(x) = \frac{u + \varepsilon_1 - 2x}{u - \varepsilon_1}; \tag{3.43}$$

this function linearly maps ε_1 to 1 and u to -1. Note the inversion of the axis: although $\varepsilon_0 < \varepsilon_1$, we have $f(\varepsilon_0) > f(\varepsilon_1) = 1$. We define the polynomial C_p as a rescaling of the Chebyshev polynomial T_p:

$$C_p(x) \coloneqq \frac{T_p(f(x))}{T_p(f(\varepsilon_0))}. \tag{3.44}$$

In this way, we are sure that $C_p(\varepsilon_0) = 1$; moreover, by definition of $f(x)$ and by equation (3.37):

$$|C_p(x)| \leq \frac{1}{T_p(f(\varepsilon_0))}, \qquad \text{for } x \in [\varepsilon_1, u]. \tag{3.45}$$

Substituting $x = f(\varepsilon_0)$ in (3.42), we justify the value of the compression $\sqrt{\Delta}$ expressed in the thesis of the theorem, because by linearity of $f(x)$:

$$\frac{f(\varepsilon_0) - 1}{f(\varepsilon_0) + 1} = \frac{f(\varepsilon_1) - f(\varepsilon_0)}{f(u) - f(\varepsilon_0)} = \frac{\varepsilon_1 - \varepsilon_0}{u - \varepsilon_0}. \tag{3.46}$$

□

We point out that an analogous construction of a precise AGSP tailored on the spectrum is possible with other polynomials. For example, even with the polynomials of the truncated Taylor expansion of $e^{-\beta x}$ that we talked about in the previous section. Anyway, because of the optimal property of Chebyshev polynomials, it will result that the Δ given by other polynomials will be higher and the compression worse.

3.6.3 MPO construction

Thanks to theorem 3.3, we decide to implement a Chebyshev kicker as MPO. Given the heavy operations needed to multiply together and compress two MPOs, we will adopt a scheme to use the minimum number of MPO·MPO products; this is why we will not adopt the recursion relation (3.36). Instead, we note that every Chebyshev polynomial has either only odd or only even powers of the argument. Then, the operator $T_p(O)$ (the Chebyshev polynomial of degree p of the MPO O) will be produced by saving the square O^2 and repeatedly multiplying I or O with it, according to p being even or odd. Finally, the powers of O will be summed, weighted by the proper Chebyshev coefficients. To know the correct number, it is necessary to implement a function $\mathrm{coeffT}(p, m)$ that returns the coefficient of the polynomial of degree p of the power m ($0 \leq m \leq p$), so for example:

$$\mathrm{coeffT}(3,3) = 4, \quad \mathrm{coeffT}(3,2) = 0, \quad \mathrm{coeffT}(3,1) = 3, \quad \mathrm{coeffT}(3,0) = 0. \tag{3.47}$$

This function, in turns, can be safely defined on the recursive relation.

The steps to generate the MPO $T_p(O)$ are described by the following algorithm:

Algorithm 3.3: Chebyshev MPO, construction

1. accept MPO O and positive integer $p \geq 0$;
2. construct the identity MPO I;
3. if $p = 0$ then return MPO $T = I$;
4. otherwise if $p = 1$ then return MPO $T = O$;
5. otherwise continue and construct MPO O^2 with MPO·MPO product;
6. compute the MPS+MPS sum

   $$P = \mathrm{coeffT}(p, p)O^2 + \mathrm{coeffT}(p, p-2)I$$

   and compress it;
7. put $m = p - 4$;
8. while $m \geq 0$ do the following steps:

(a) compute the MPO·MPO product $Q = O^2 \cdot P$ (Q is only a temporary MPO to save the result);

(b) compute the MPS+MPS sum $P = Q + \text{coeffT}(p, m)I$;

(c) perform an SVD compression on P;

(d) return to point 8 with $m \leftarrow m - 2$;

9. if p is even return $T = P$;

10. otherwise return the MPO·MPO product $T = P \cdot O$ compressed.

To obtain the complete Chebyshev kicker $C_p(H)$, we have to pass to the above algorithm an MPO O corresponding to the appropriately "shifted" hamiltonian by function $f(x)$ of theorem 3.3. We see that $f(H)$ is simply:

$$f(H) = \frac{u + \varepsilon_1}{u - \varepsilon_1} I - \frac{2}{u - \varepsilon_1} H. \tag{3.48}$$

Afterwards we have to rescale $T_p(f(H))$ by the number $1/T_p(f(\varepsilon_0))$ and we have completed the kicker.

It is in the last passage that the information about the spectrum is required. Again, this might be a disadvantage for the kicker, since we usually search ground states of systems we know almost nothing about, apart the hamiltonian. A convenient time to use such kicker could be when the hamiltonian is a smooth function of its parameters and we investigate the ground state changing slowly those parameters. Suppose that, for the first set of parameters, we start to look for the ground state with the hamiltonian kicker, that does not need any spectrum-related variable. Then we would know the ground state energy for those parameters and, for the next very close set, we could adjust ε_0 near that value.

Chapter 4

Results

This chapter is dedicated to the presentation of the tests carried out on the new algorithm and their results.

All the tests are performed on the Ising model, because it is a simple and completely solvable model, so that we can extrapolate the most about advantages and disadvantages of the kicker strategy. Moreover, the physics of the Ising model is intuitive, but it also presents interesting behaviour and a quantum phase transition. We start with a little introduction to the model and to its quantum phase transition.

We proceed confronting the ground state energies found by the hamiltonian kicker and by DMRG with the theoretical values, changing the coupling constants of the model. This will determine that the new algorithm has been correctly implemented.

Since the imaginary time evolution kicker and the Chebyshev kicker are controlled by some parameters, we try to understand how the speed of energy descent varies, changing those parameters. Afterwards, we compare the various kicker together and with DMRG.

We will find that, with very long spin chains, the kicker strategy is initially faster than DMRG in the energy descent. We will try to express quantitatively this velocity, plotting the computational time required by the various methods to reach a fixed fraction of the ground state energy in function of the number of sites of the system.

Finally, we give the results of a hybrid technique that starts with the hamiltonian kicker and switches to DMRG after a fixed number of cycles.

4.1 The Ising model

The Ising model has spin-1/2 particles for every site; this means that the local Hilbert space dimension is $d = 2$. The interaction is between nearest-neighbor sites and there is also an *external field* acting on the single sites separately. It is very important to notice that the external field and the nearest-neighbor interaction have orthogonal direction. By this we mean: the energy in the external field depends on the components of the spins along the z-direction, while the nearest-neighbor interaction depends on the alignment along the x-direction.

In one dimension, the hamiltonian is:

$$H = J \sum_{s=1}^{N-1} S_s^x S_{s+1}^x + h \sum_{s=1}^{N} S_s^z, \tag{4.1}$$

where S_s^x and S_s^z are the generators of the spin-1/2 Lie algebra representation of the angular momentum in direction x and z; they act on site s. They are a half of the corresponding Pauli matrices:

$$S^{x,z} = \frac{1}{2}\sigma^{x,z}, \qquad \sigma^x = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \qquad \sigma^z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}. \tag{4.2}$$

In section 2.13, we already saw how to translate a hamiltonian like (4.1) in MPO form.

The Ising hamiltonian has two coupling constants: h that regulates the strength of the external field and J that determines the intensity of the nearest-neighbor interaction. The physics of the system, and of the ground state in particular, depends on the value of these two parameters.

We will not give derivations of the theoretical results; they can be found in the original paper [62] or in Sachdev's book [5]. What happens for the ground state is that, when the nearest-neighbor interaction is negligible compared to the transverse field, $h \gg J$, the spins of the lattice are aligned along the z-direction and, in the thermodynamic limit, there is no magnetization along x (i.e. they are eigenvectors of S^z). This phase is called *quantum paramagnet*. On the contrary, when $h \ll J$ is negligible or null, and $J < 0$, all the spins are directed along x, with a non-null magnetization along x. This phase is *magnetically ordered*. In between these two choises of coupling constants, the systems go through a *quantum phase transition*, namely a phase transition that happens at zero temperature, due to quantum fluctuations. In this case, it is the external transverse field that allow the magnetically ordered spins to do quantum tunnelling to a state parallel to the field. It is possible to prove that the critical field that mark the change of phase is $|h/J| = 0.5$.

4.2 Ground state energy

As a first test, we want to determine if the implementation of the kicker strategy is reliable and the ground state energies found are physically correct. The theoretical expression for the ground state energy ε_0, in function of the coupling constants of the model, has been found in [62]:

$$-\frac{\varepsilon_0}{JN} = \frac{1}{\pi}\left(\frac{h}{J}+\frac{1}{2}\right) E\left(\sqrt{\frac{2(h/J)}{(h/J+1/2)^2}}\right), \tag{4.3}$$

where $E(x)$ is the elliptic integral of the second kind. We can see that the ground state energy depends on the coupling constants only through their ratio. In other words, the value of J sets a scale for the energy. We decided, then, to fix the value of $|J| = 1$ and to let only h vary. The ground state energy is independent from the sign of J.

The results of the ground state energies computed with the hamiltonian kicker and with DMRG are compared with the theoretical line in figure 4.1. We simulated the Ising chain with $N = 120$ sites and, here, we plot the values normalized by N in function of h/J.

The reason for a high number of spins is that the theoretical result was obtained for periodic boundary conditions on the chain. Algorithm working with MPSs are slower with PBC and prefer the open boundary conditions. In the OBC case, the number of bonds between sites is down by 1 with respect to PBC (see section 2.1). It is easy to understand from hamiltonian (4.1) that, when $h = 0$, the ground state energy is proportional to the number of nearest neighbors, namely the number of bonds. This is why, for a small number of sites N and in a simulation with OBC, we would see a difference with the theory of PBC at $h = 0$[1]. Choosing N very large, the difference of 1 bond is less and less important and, indeed, it is barely visible in figure 4.1 for low h. Apart for this very small discrepancy, that we already accounted for, there is perfect agreement between theory and simulation (for both DMRG and hamiltonian kicker) and we can safely assume that the kicker strategy was correctly implemented.

[1]For $h \gg J$, instead, we find no difference because the ground state energy is proportional to the number of sites (put $J = 0$ in (4.1))

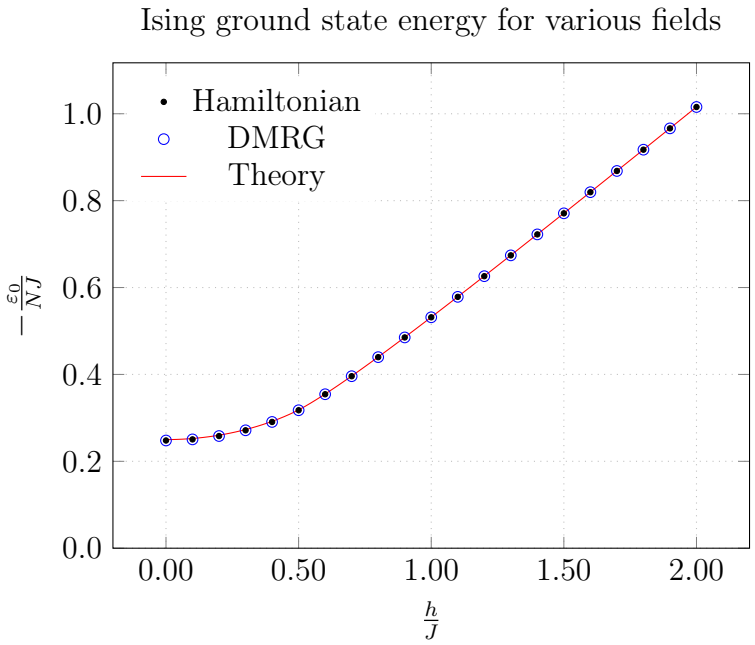


Figure 4.1: The ground state energies of the Ising model at different values of the transverse field as found with the hamiltonian kicker and with DMRG. Results from both methods are compatible between them and with the theoretical line. The theoretical result is the plot of equation (4.3) with respect to the ratio of the field h and of the nearest-neighbor coupling J.

4.3 Comparison with imaginary time evolution and Chebyshev kicker

Now that we know that the new method is reliable, we can go on to test whether there is or not any advantage in using more sophisticated kickers than the hamiltonian one. In particular, we want to know if imaginary time evolution (ITE) kickers or Chebyshev kickers (CK) are faster and how their speed change varying the parameters that define them.

At every kick, the program of the kicker strategy returns the energy of the current approximation of the ground state. To measure the velocity of the energy descent, we opted to plot these energy values with respect to the computational time. The computational time that appears in the graphs corresponds to the physical number of seconds that the machine employed to arrive at that energy. We used the same *reference machine* for all the calculations, to maintain the compatibility across the results. The machine was a cluster of 24 cores and each core was an Intel® Xeon® X5675 with 3.07GHz of clock speed and 12MB of cache.

In the graphs, it is sometimes possible to note that series of energy values stop before others. This is due to the choice of stopping mechanism: we monitored the convergence of energy (E_1) as described at the end of section 3.2.2. The threshold energy difference chosen was $\epsilon = 10^{-5}$ with $t = 10$; this means that when the energy difference, with the previous step, was lower than ϵ for t times the program would stop its course.

We want to make clear that what follows is just an exploratory data analysis, where we address some basic questions in a few specific cases. The number of variables and models, to which such analysis should be applied, is too high to reach conclusive results.

4.3.1 Imaginary time evolution kicker

To test the imaginary time evolution kicker, we chose a system with $N = 80$ spins whose state was approximated by an MPS of bond link $D = 20$. The field strength was set to $h = 0.8$; the choice of this parameter is quite arbitrary, but we did not want a value too near to 0, where the ground state is trivial, nor too near to 0.5, where the system is critical and the gap null.

As we know from section 3.5, the ITE kicker depends on two parameters, p and β. We put $p = 3$ to have a good approximation of the exponential without too much computational cost, but we let β vary. Equation (3.30) tells us that the range of good approximation depends on the value of β; anyway, usually, we do not know the interval of the spectrum, thus, we do not know

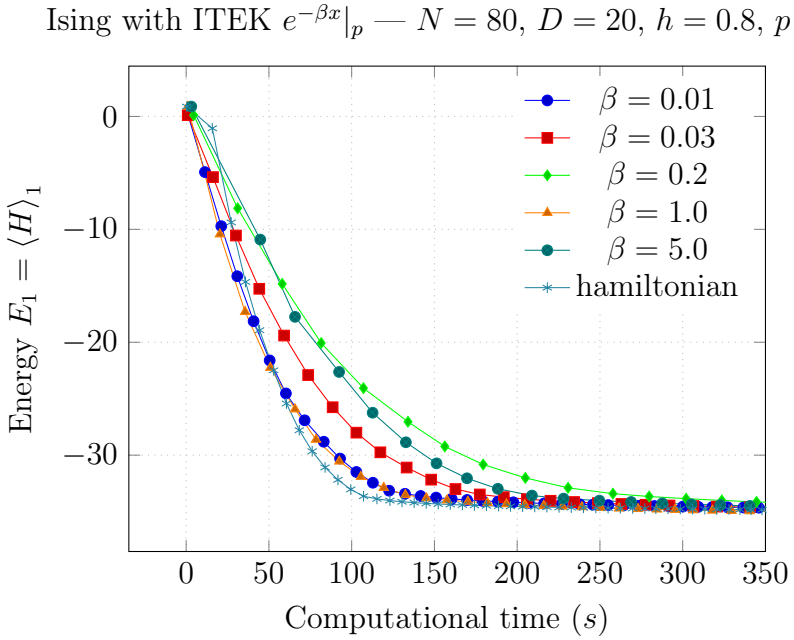


Figure 4.2: Energy descent plotted against computational time for ITE kicker at various β.

how β should be chosen, nor if it is important that the good approximation fits the spectrum. It results that the ground state energy for $h = 0.8$ and $N = 80$ is about -35.16, therefore, the spectrum for those parameters is included in $[-35.16, 35.16]$ (because the Ising spectrum is symmetrical with respect to the origin). Using this interval as range of validity for the approximation and inverting (3.30) for a 10% error ($\epsilon = 0.1$), we have $\beta \approx 0.0295$. We tried to understand which is the optimal choice of β: whether we should underestimate it ($\beta = 0.01$), overestimate it ($\beta = 0.2$, plus other values) or choosing it fitted to the spectrum ($\beta = 0.03$). The results shown in figure 4.2 indicate that, among these three values, a small $\beta = 0.01$ results in an initially faster descent, but it turns out that $\beta = 1$ is better. Even the convergence of $\beta = 1$ is earlier, as it can be see from the zoom in figure 4.3. No improvement derives from choosing a higher β. We can also see how $\beta = 1$ presents similar initial performance as the hamiltonian kicker (shown in both figures), but, still, better convergence. We deduce that β in the order of unity is optimal in this case.

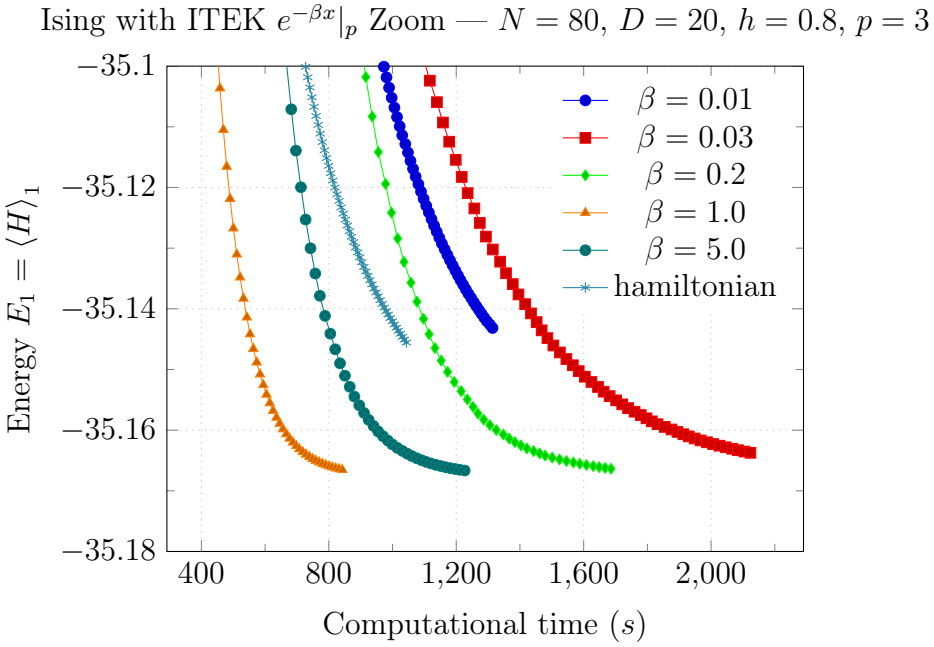


Figure 4.3: Detail of the energy descent for ITE kicker to show the convergence of energy.

4.3.2 Chebyshev kicker

About Chebyshev kicker, we can do a very similar analysis for the knowledge of the spectrum, but there is an additional problem concerning the choice of the gap. Thus, first of all, we fix the extremes of the spectrum to $\varepsilon_0 = -35.16$ and $u = 35.16$, then, we let the gap $g = \varepsilon_1 - \varepsilon_0$ vary with the following values: 8, 17, 26, 35, 43, 52, 61, that roughly correspond to the eighths of the spectrum interval. Results in figure 4.4 show that the choice of gap does not influence significantly the convergence of energy, although, from figure 4.5, we see that the simulations with smaller gaps ($g = 8$, $g = 17$) are those that stopped before at the right value.

Regarding the size of the spectrum interval, we plot in figure 4.6 three different simulations with $[\varepsilon_0, u] = [-18, 18]$, $[\varepsilon_0, u] = [-35.16, 35.16]$ and $[\varepsilon_0, u] = [-70, 70]$. These spectrum intervals correspond to a half, the correct size and the double of the real spectrum, respectively. We plot only the zoom for the final convergence since it is the most interesting part. In the graph, we also add the hamiltonian kicker performance for comparison. The choice of the gap, given the above results, was about a quarter of the spectrum interval length, so $g = 9$, 17, 35. We can see how the convergence is very similar, but, apparently, it is better to take a spectrum of the correct size

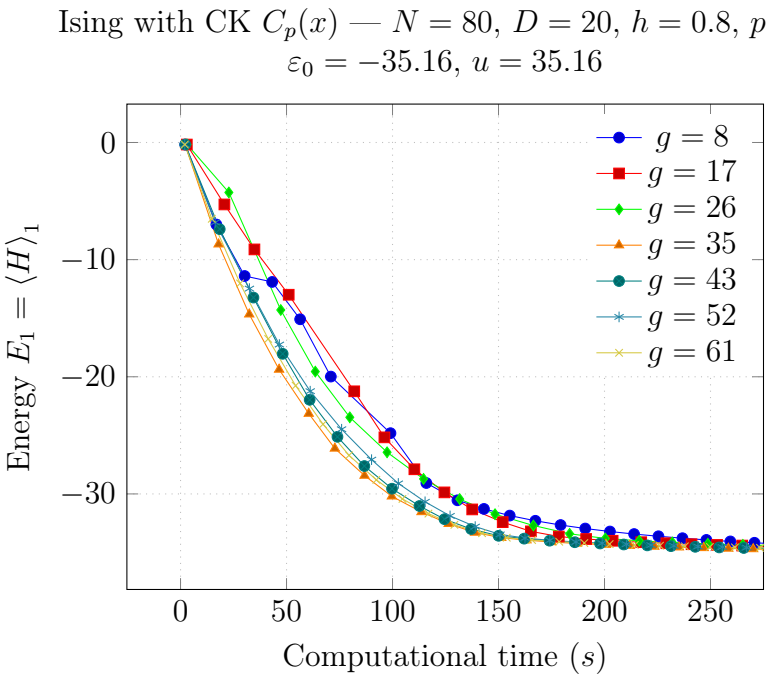


Figure 4.4: Energy descent of the Chebyshev kicker with spectrum interval fitted to the real value and the gap g varying.

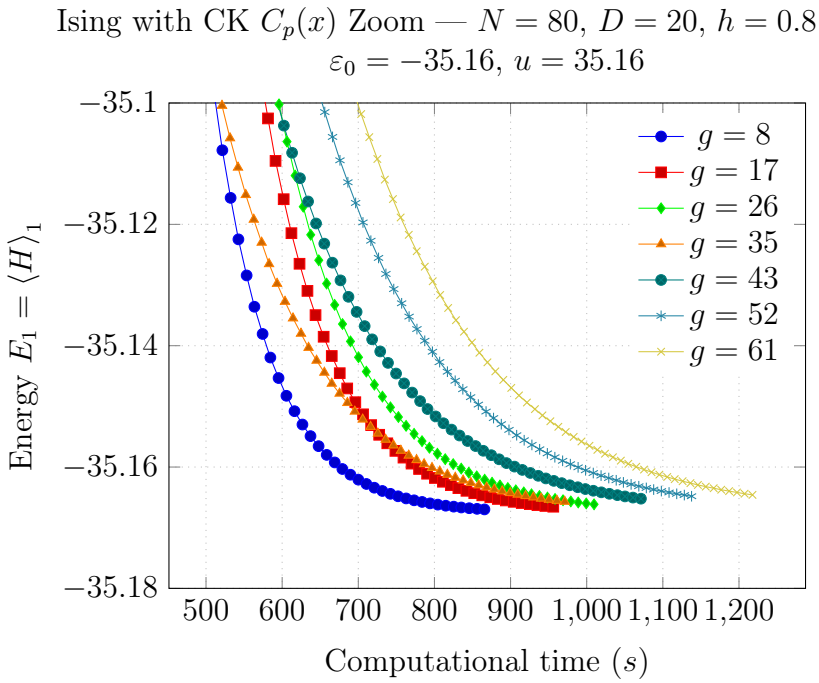


Figure 4.5: Detail of the energy descent of the Chebyshev kicker with varying gap to highlight the convergence.

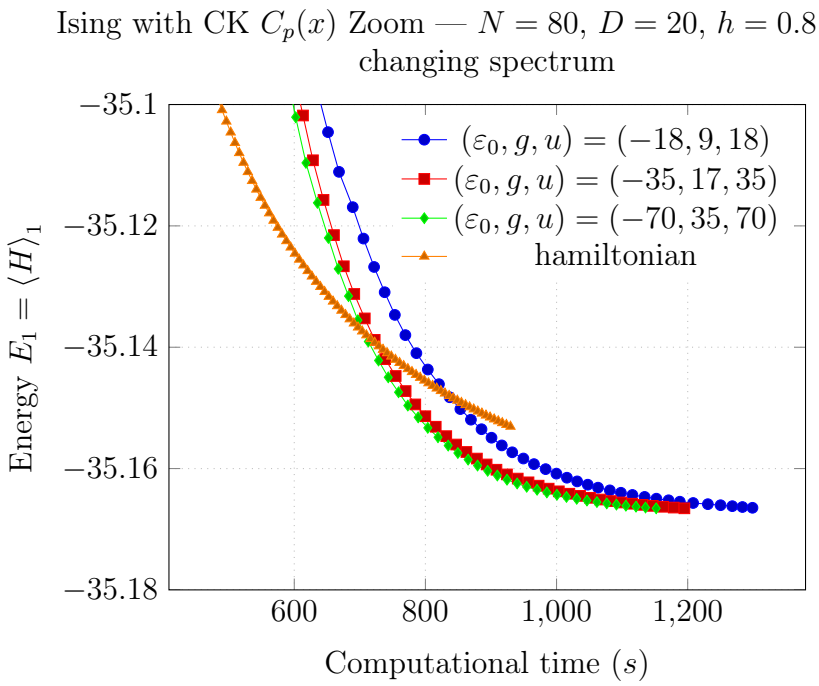


Figure 4.6: Energy descent for the Chebyshev kicker using different spectra and compared to the hamiltonian kicker.

or wider. Moreover the hamiltonian kicker result slower than the Chebyshev kicker.

4.4 Comparison with hamiltonian kicker and DMRG

We decided to compare the density matrix renormalization group (section 3.1) to the performance of the hamiltonian kicker. The main reason is that the other kickers need knowledge of the spectrum, that we often do not possess, hence, we put ourselves in the worst case scenario. We tested thoroughly the energy descents of both these methods at various lengths, fields and bond link dimensions. Let us have a look at a simple graph, like that in figure 4.7, to begin with. That simulation involved a system with $N = 40$ spin, field $h = 0.5$ and the MPS had a bond link dimension of $D = 20$. Numerically, this kind of simulation is not very demanding. A few comments on DMRG: the descent have an almost constant slope globally and the energy go down regularly at every step, until it reaches the correct ground state eigenvalue

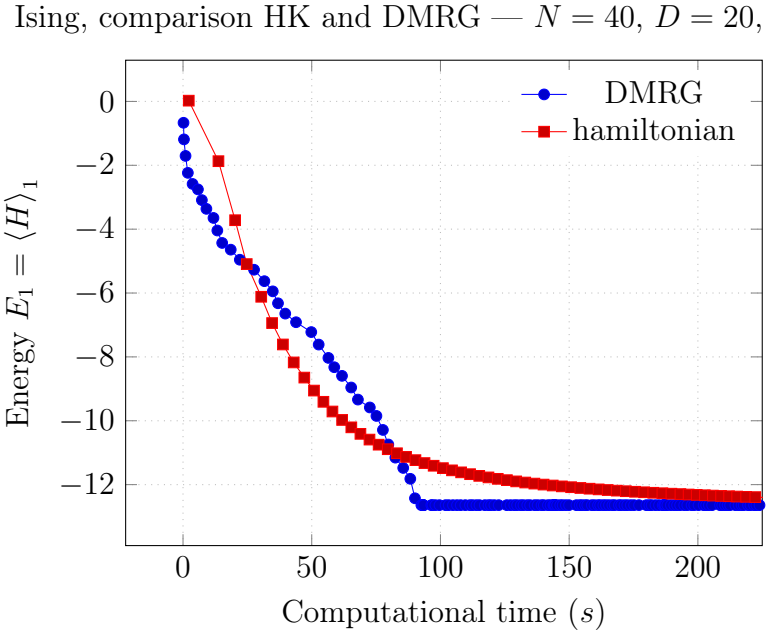


Figure 4.7: Comparison of the energy descents of DMRG and hamiltonian kicker for a small simulation.

(final plateau). Counting, from the data, the number of steps to reach the ground state energy, we saw that it was equal to the number of sites. This is due to the fact that the DMRG algorithm updates every site with an iterative eigensolver, like the Jacobi-Davidson method, to find the matrices that minimize the energy. Starting from a random vector, it needs to pass through all the chain before reaching the ground state, because all the elements of the matrices have to be changed. The behaviour of the hamiltonian kicker, instead, is very similar to that predicted in section 3.3 and very different from DMRG. Here, every matrix of the MPS is substituted by a new one at every step[2] and the energy descent is initially fast and then it slows down, reaching the ground state energy only after a relatively long time and with lower precision.

Consider, now, a bigger simulation, with $N = 120$ and $D = 50$ (the field still $h = 0.5$); we represent it in figure 4.8. In this situation, the constant slope of DMRG is much clearer. Moreover, the initially very fast energy descent of the hamiltonian kicker is very neat too. Indeed, the hamiltonian kicker reaches a decent approximation of the ground state energy thousands

[2]This is also one of the reasons why we cannot compare DMRG and kicker strategy through the number of steps, but only with computational time.

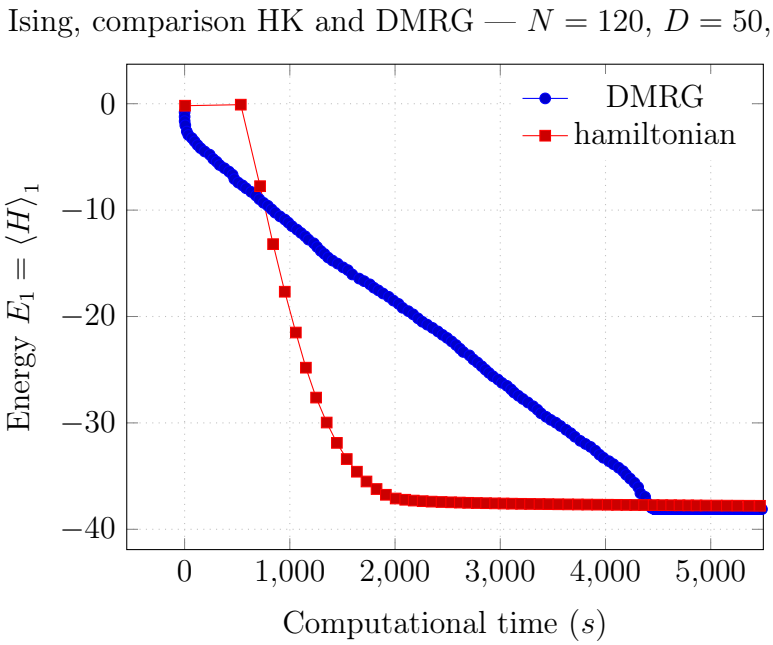


Figure 4.8: Comparison of the energy descents of DMRG and hamiltonian kicker for a large simulation

of seconds before the DRMG. You may notice that the very first kick of the hamiltonian produces a state with the same energy and only with ensuing kicks the energy begin to going down. This happens in other situations as well; we suppose this is due to an "adjustment kick", needed to bring the first random vector to point in the right direction. Anyway, the important part is that we believe that DMRG requires a computational time linear with the size of the system *to reach a good approximation ground state*, while the kicker strategy, and, in particular, the hamiltonian kicker, demands a lower power of N.

In order to give some foundations to the above conjecture, we computed the computational times require by the two algorithms to reach the 95% of the true ground state energy with different lengths, both for $D = 20$ (figure 4.9) and for $D = 50$ (figure 4.10). The field strength is still $h = 0.5$: the kicker strategy has a better performance over DMRG at critical field. For the two bond link dimensions, we see a linear growth in the computational time required by DMRG as the system size increases, as it was suggested by the previous graphs. In the case of $D = 20$, the hamiltonian kicker is slower until $N = 60$, but in $D = 50$ case it is always faster. Moreover, although there is an augment in the computational time demanded by the kicker for

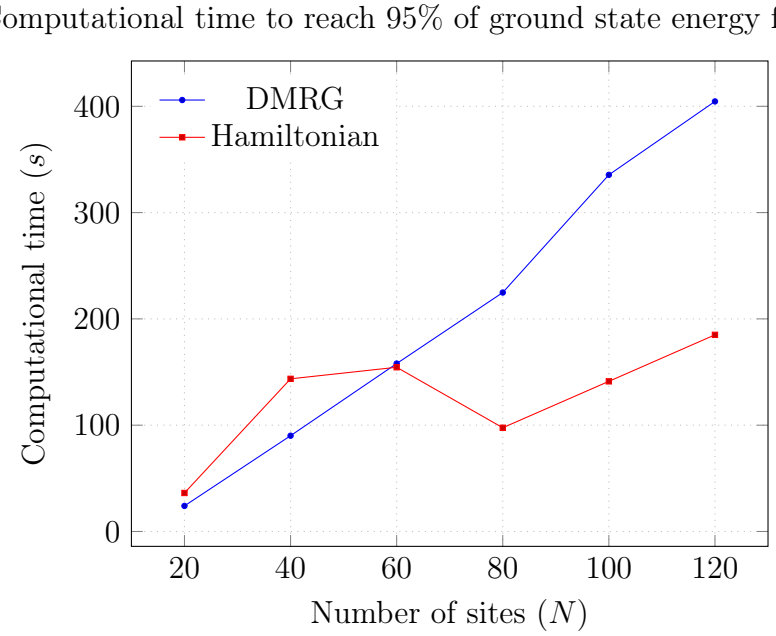


Figure 4.9: The computational time needed to reach at least 95% of the ground state energy plotted against an increasing number of site: 20, 40, 60, 80, 100, 120; for $D = 20$.

bigger systems, the rate is lower than linear.

4.5 Hybrid technique

The results of the previous section suggest the combined use of both kicker strategy and DMRG: the former is initially very fast, but slow at the end; the latter quickly finds the correct eigenvalue of the ground state, when it is near to it. We expect that, feeding the DMRG with a state sufficiently close to the ground state energy, we obtain a speedup for the algorithm and, if we use the hamiltonian kicker to reach that state faster, a global speedup.

However, when we try this *hybrid technique* for $N = 120$, we obtain the results shown in figures 4.11 and 4.12, for $D = 20$ and $D = 50$, respectively. We do observe an improvement with respect to the kicker alone, but not with respect to DMRG alone. At the moment when we switch algorithm, we do not see a steep fall towards ground state, but a slow, regular descent as that of DMRG alone, except that it is shrunk and shifted. The time difference between switch and convergence to ground state is compatible to the time required by DMRG alone to reach ground state. Mainly for this reason, we

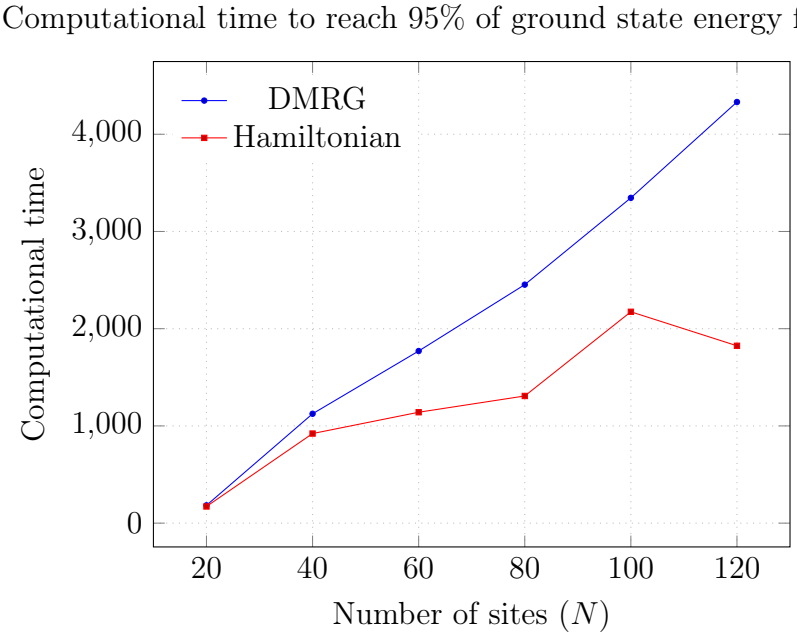


Figure 4.10: The computational time needed to reach at least 95% of the ground state energy plotted against an increasing number of site: 20, 40, 60, 80, 100, 120; for $D = 20$.

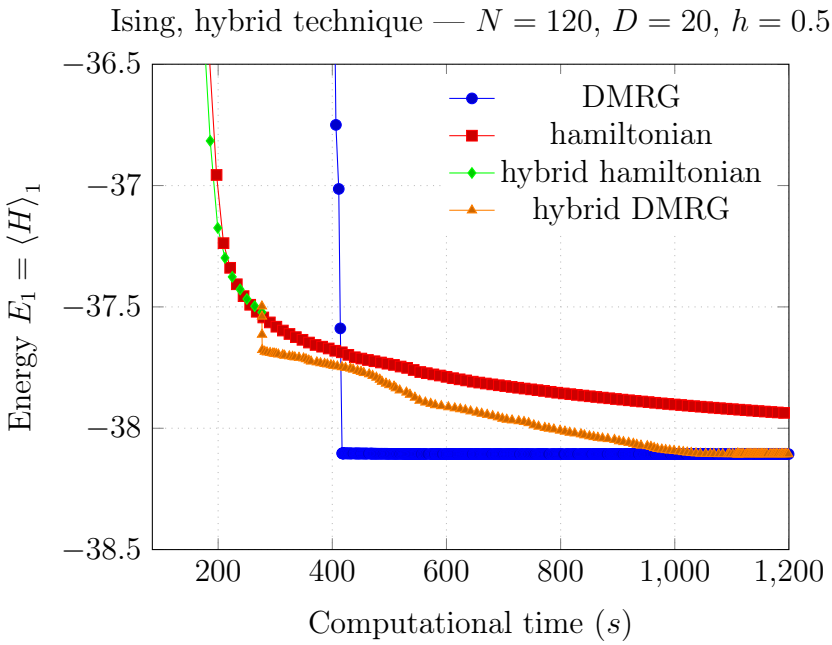


Figure 4.11: Energy descent of hybrid technique compared to hamiltonian kicker alone and DMRG alone. We split the hamiltonian and DMRG part of the hybrid algorithm to easily spot the point where the algorithm switch.

believe that DMRG must undergo the same process of matrix manipulation for every site and it does not matter if the energy of the state is already very low. This behaviour could be influenced, once again, by the iterative eigensolver used: if the eigensolver encounters a matrix not in a suitable form, it takes a lot of time to find its lower eigenvectors. Another, possible explanation could be that the DMRG program expects the matrices of the vector in a particular gauge, while the state fed do not possess such gauge. Anyway, further research is demanded to understand if these difficulties can be avoided and an efficient hybrid technique implemented.

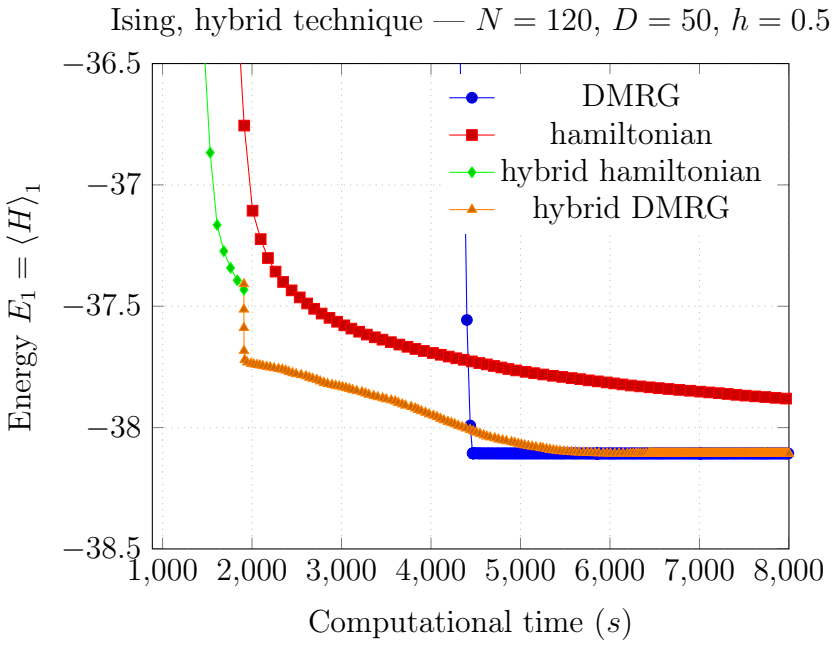


Figure 4.12: Energy descent of hybrid technique compared to hamiltonian kicker alone and DMRG alone. We split the hamiltonian and DMRG part of the hybrid algorithm to easily spot the point where the algorithm switch.

Conclusions

To conclude this work we now want to summarize the most important steps we went through and comment the results we obtained.

We started out this work by showing some of the results of quantum information theory, in order to understand the full meaning of the area laws for quantum entanglement in many-body systems. These area laws say that, in ground states of local, gapped hamiltonians, the entropy of a subregion does not follow the most probable behaviour: increasing with the size of the subregion; on the contrary, the entropy grows proportional to the size of the boundary of the subregion. In one dimension, this means that the entropy is bounded by a constant for any size of the subregion, which leads to major simplifications in the ground state. Now, we know that the ground state resides in a very small class of states.

We went on reviewing the theory of Matrix Product States that represent a parametrization of such class of states with area law. After their definition and some examples, we explained their gauge properties and how they could be derived from general terms. It results that every many-body state can be written in MPS form, i.e. as the product of many matrices that compute the coefficients of the state, but, usually, the size of the matrices diverges exponentially with the system size. To obtain an area law state, it is necessary to force a cutoff to the dimensions of the matrices, independent from the number of particles.

Many more or less specific and technically involved operations followed; these were all needed for the construction of the new algorithm presented in this work. For a quick summary, they were: overlaps, expectation values, transfer operators, sum and compressions, to deal with the increasing size of matrices that derives from the sum.

Afterwards, we talked about the generalization of MPS to operators. This formalism is very important because it preserves the MPS structure, so, we decided to express every operator, like the hamiltonian, in this way.

Before continuing with the new algorithm, we stopped for a moment to introduce the Density Matrix Renormalisation Group. It is an established

method that search ground states variationally on the class of MPS and we used it to compare its performance with that of our new procedure.

At this point, we presented the original contributions of this work: first of all, the idea of a new process for ground state search, that consists in repeatedly compute a new vector from an old one applying an operator, called kicker, then minimize the energy in the bi-dimensional subspace spanned by these two vectors and restart from the vector with the minimum energy. The second contribution was the evaluation of the technical details concerning the correct and efficient implementation of this new algorithm. After which came the analysis of the computational complexity and, then, the extensive theoretical study of some very important kickers and the test of the new strategy for the Ising model.

About the kickers, we considered the hamiltonian, at first, and we found out that it is very versatile, because it has a very small size, but the convergence towards ground state is always asymptotic. Then, using the spectrum decomposition, we explored the properties of two operators that define approximate ground state projectors and more efficiently suppress higher spectrum components. These are the imaginary time evolution kicker and the Chebyshev kicker. They both need parameters that require knowledge of the spectrum; this may be a problem sometimes, but it could be exploited with many similar simulations, e.g. when the parameters change only slightly.

After the theoretical study of the kicker, we carried out numerical tests with them. All the tests were performed with the Ising model, briefly presented before. We proved the correct implementation of the algorithm, comparing the ground states found by the hamiltonian kicker to the theoretical values, varing the external field in the model. Then, we checked the performance of imaginary time evolution kicker and Chebyshev kicker varying their parameters, but we found out that their response is very problem specific. The next step was to compare the hamiltonian kicker with DMRG and it was clear that the hamiltonian kicker had an initial advantage over DMRG for system with a lot of spins. In the end, we tried to combine kicker strategy and DMRG with a hybrid technique, but we could not find any significant advantage of this method, due probably to the interface between the two programs used.

Finally, we want to answer the question on how this work could continue. It surely seems important to investigate whether it is possible, and which is the best method, to interface the kicker strategy with DMRG, in order to take advantage of the great initial speed of the first. In case this could not work, another useful course of action would be to explore whether, for higher threshold than 95% of the ground state energy, there are systems where the kicker strategy is faster than DMRG. In that case, we could argue

that, for very big systems, the kicker strategy reaches a sufficiently accurate approximation of the ground state before DMRG. There is also the need to extend the test to other ranges of parameters not explored here and to other kinds of models, like disordered-systems or with longer-than-nearest-neighbour interactions.

Appendix A

Singular Value Decomposition

The *Singular Value Decomposition* (SVD) is a very general factorization that we borrow from linear algebra [63]. It states that every matrix M (real or complex and even rectangular) is decomposable in other three matrices, where the first and the last are isometries (U and V) and the middle one is diagonal (Σ):

$$M = U\Sigma V^\dagger, \qquad m_{ij} = \sum_k u_{ik}\sigma_k v^\dagger_{kj}. \tag{A.1}$$

The two following conditions are always guaranteed for U and V:

$$U^\dagger U = \mathbb{1}, \quad V^\dagger V = \mathbb{1}. \tag{A.2}$$

The columns of U contain the *left singular vectors* of M, while the columns of V (not adjoint) contain the *right singular vectors* of M. The diagonal elements of Σ are always real, non-negative and listed in decreasing order and they are called *singular values*.

SVD comes in different forms (or conventions), depending on how the matrix multiplications are respected. Suppose matrix M has dimensions $m \times n$:

- *Full SVD*: U is unitary with dimensions $m \times m$ and V is unitary with dimensions $n \times n$, so it is also true that $UU^\dagger = \mathbb{1}$ and $VV^\dagger = \mathbb{1}$. Matrix Σ is rectangular $m \times n$ with only $\min(m, n)$ non-zero elements on the diagonal.

- *Reduced SVD*: from the "full" form we observe that some rows or columns of Σ are zero-vectors; it is possible to shrink Σ so that it assumes a square shape with $\min(m, n)$ rows (or columns), thus reducing the columns of U or the rows of V:

$$\text{if } m > n, \quad U\Sigma = \begin{pmatrix} U_1 & U_2 \end{pmatrix} \begin{pmatrix} \Sigma_1 \\ 0 \end{pmatrix} = U_1\Sigma_1, \tag{A.3}$$

	Full	Reduced	Minimal
U	$m \times m$	$m \times \min(m,n)$	$m \times r$
Λ	$m \times n$	$\min(m,n) \times \min(m,n)$	$r \times r$
V	$n \times n$	$\min(m,n) \times n$	$r \times n$

Table A.1: dimensions of SVD matrices, generated from M of dimensions $m \times n$, in the different forms; r is the rank of M.

$$\text{if } m < n, \quad \Sigma V = \begin{pmatrix} \Sigma_1 & 0 \end{pmatrix} \begin{pmatrix} V_1 \\ V_2 \end{pmatrix} = \Sigma_1 V_1. \tag{A.4}$$

Then, if $m > n$, U will be just an isometry and V will be unitary, viceversa when $m < n$. The dimensions of U and V are easily deducible and they are listed in table A.1.

- *Minimal SVD*: pushing further this process, there might be some singular values equal to zero. Suppose that at least one singular value is zero and let $r < \min(m,n)$ be the number of non-zero singular values, then we can shrink again the matrices. U and V will only be isometries and never unitaries, with dimensions $m \times r$ and $r \times n$ respectively; Σ is $r \times r$ diagonal.

To understand what is the meaning of SVD consider the reduced form. Let $p = \min(m,n)$, the columns of U and V contain the p right and left singular vectors (respectively):

$$U = \begin{pmatrix} \mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_p \end{pmatrix} \qquad V = \begin{pmatrix} \mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_p \end{pmatrix} \tag{A.5}$$

and we have $\Sigma = \operatorname{diag}(\sigma_1, \sigma_2, \ldots, \sigma_p)$, with $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_p \geq 0$. It is easy to prove the following relations:

$$M\mathbf{v}_i = \sigma_i \mathbf{u}_i, \qquad M^\dagger \mathbf{u}_i = \sigma_i \mathbf{v}_i. \tag{A.6}$$

The number of non-zero singular values r is also the rank of the matrix: $\operatorname{rank}(M) = r \leq p$. From (A.6) we can expand matrix M as a sum of simpler matrices coming from the outer product of two corresponding singular vectors, weighted with the singular values:

$$M = \sum_{k=1}^{r} \sigma_k \mathbf{u}_k \mathbf{v}_k^\dagger. \tag{A.7}$$

From M we can extract a group of matrices with rank r' less than r:

$$M_{r'} = \sum_{k=1}^{r'} \sigma_k \mathbf{u}_k \mathbf{v}_k^\dagger, \qquad r' \leq r. \tag{A.8}$$

Note that SVD coincides with eigenvalue decomposition only when matrix M is positive. If M is just hermitian then the singular values are the absolute value of the eigenvalues; for SVD the sign is incorporated in one of the corresponding singular vectors. The eigenvalue decomposition requires that the unitary matrices are one the adjoint of the other; on the other hand the singular values of SVD must always be non-negative.

A remarkable theorem confirms the importance of SVD even when approximations are involved. Suppose we want to reproduce as closely as possible matrix M of rank r with matrix $M_{r'}$ of rank $r' \leq r$. The nearest matrix, in the Frobenius distance, is obtained retaining the r' greatest singular values of M and the corresponding singular vectors. The Frobenius distance is induced by the Frobenius norm:

$$\|M\|_F^2 = \sum_{i,j} |m_{ij}|^2 . \tag{A.9}$$

Theorem A.1 (Eckart-Young low-rank approximation)**.** *Let M be a matrix of rank r. The nearest matrix to M in the Frobenius distance that has rank $r' \leq r$ is given by $M_{r'}$:*

$$\min_{\mathrm{rank}(N)\leq r'} \|M - N\|_F = \|M - M_{r'}\|_F = \sqrt{\sum_{k=r'+1}^{r} \sigma_k^2}. \tag{A.10}$$

Bibliography

1. R. P. Feynman, "Simulating physics with computers", *Int. J. Theor. Phys.* **21**, 467–488 (1982).
2. I. Bloch, J. Dalibard, W. Zwerger, "Many-body physics with ultracold gases", *Rev. Mod. Phys.* **80**, 885–964 (2008).
3. W. Heisenberg, *Z. Phys.* **38**, 441 (1926).
4. E. Lieb, T. Schultz, D. Mattis, "Two soluble models of an antiferromagnetic chain", *Annals of Physics* **16**, 407–466 (1961).
5. S. Sachdev, *Quantum phase transitions* (Cambridge University Press, Cambridge ; New York, Second edition, 2011), 501 pp.
6. H. Bethe, "Zur Theorie der Metalle", *Z. Physik* **71**, 205–226 (1931).
7. K. G. Wilson, "The renormalization group: Critical phenomena and the Kondo problem", *Rev. Mod. Phys.* **47**, 773–840 (1975).
8. M. Troyer, U.-J. Wiese, "Computational Complexity and Fundamental Limitations to Fermionic Quantum Monte Carlo Simulations", *Phys. Rev. Lett.* **94**, 170201 (2005).
9. S. Östlund, S. Rommer, "Thermodynamic Limit of Density Matrix Renormalization", *Phys. Rev. Lett.* **75**, 3537–3540 (1995).
10. J. Dukelsky, M. A. Martín-Delgado, T. Nishino, G. Sierra, "Equivalence of the variational matrix product method and the density matrix renormalization group applied to spin chains", **43**, 457–462 (1998).
11. G. Vidal, J. I. Latorre, E. Rico, A. Kitaev, "Entanglement in Quantum Critical Phenomena", *Phys. Rev. Lett.* **90**, 227902 (2003).
12. J. Eisert, M. Cramer, M. B. Plenio, "*Colloquium*: Area laws for the entanglement entropy", *Rev. Mod. Phys.* **82**, 277–306 (2010).
13. G. Vidal, "Efficient Simulation of One-Dimensional Quantum Many-Body Systems", *Phys. Rev. Lett.* **93**, 040502 (2004).

14. F. Verstraete, J. J. García-Ripoll, J. I. Cirac, "Matrix Product Density Operators: Simulation of Finite-Temperature and Dissipative Systems", *Phys. Rev. Lett.* **93**, 207204 (2004).

15. U. Schollwöck, "The density-matrix renormalization group in the age of matrix product states", *Annals of Physics*, January 2011 Special Issue **326**, 96–192 (2011).

16. C. Lanczos, "An iteration method for the solution of the eigenvalue problem of linear differential and integral operators", *J. Res. Nat'l Bur. Std.* **45**, 225–282 (1950).

17. I. Arad, A. Kitaev, Z. Landau, U. Vazirani, "An area law and sub-exponential algorithm for 1D systems", arXiv: `1301.1162` (2013).

18. J. von Neumann, *Mathematische Grundlagen der Quantenmechanik* (Springer, 2. Aufl. 1996, 1996), 262 pp.

19. A. Einstein, B. Podolsky, N. Rosen, "Can Quantum-Mechanical Description of Physical Reality Be Considered Complete?", *Phys. Rev.* **47**, 777–780 (1935).

20. E. Schrödinger, "Die gegenwärtige Situation in der Quantenmechanik", *Naturwissenschaften* **23**, 807–812 (1935).

21. A. K. Ekert, "Quantum cryptography based on Bell's theorem", *Phys. Rev. Lett.* **67**, 661–663 (1991).

22. C. H. Bennett, S. J. Wiesner, "Communication via one- and two-particle operators on Einstein-Podolsky-Rosen states", *Phys. Rev. Lett.* **69**, 2881–2884 (1992).

23. C. H. Bennett, G. Brassard, C. Crépeau, R. Jozsa, A. Peres, W. K. Wootters, "Teleporting an unknown quantum state via dual classical and Einstein-Podolsky-Rosen channels", *Phys. Rev. Lett.* **70**, 1895–1899 (1993).

24. M. A. Nielsen, I. L. Chuang, *Quantum Computation and Quantum Information: 10th Anniversary Edition* (Cambridge University Press, 10 Anv edizione, 2010), 702 pp.

25. R. Horodecki, P. Horodecki, M. Horodecki, K. Horodecki, "Quantum entanglement", *Rev. Mod. Phys.* **81**, 865–942 (2009).

26. M. B. Plenio, S. Virmani, "An introduction to entanglement measures", *Quant. Inf. Comp.* **7**, 1–51 (2007).

27. R. F. Werner, "Quantum states with Einstein-Podolsky-Rosen correlations admitting a hidden-variable model", *Phys. Rev. A* **40**, 4277–4281 (1989).

28. L. Masanes, "All Bipartite Entangled States Are Useful for Information Processing", *Phys. Rev. Lett.* **96**, 150501 (2006).

29. J. D. Bekenstein, "Black Holes and Entropy", *Phys. Rev. D* **7**, 2333–2346 (1973).

30. S. W. Hawking, "Black hole explosions?", *Nature* **248**, 30–31 (1974).

31. L. Bombelli, R. K. Koul, J. Lee, R. D. Sorkin, "Quantum source of entropy for black holes", *Phys. Rev. D* **34**, 373–383 (1986).

32. M. Srednicki, "Entropy and area", *Phys. Rev. Lett.* **71**, 666–669 (1993).

33. C. Callan, F. Wilczek, "On geometric entropy", *Physics Letters B* **333**, 55–61 (1994).

34. T. M. Fiola, J. Preskill, A. Strominger, S. P. Trivedi, "Black hole thermodynamics and information loss in two dimensions", *Phys. Rev. D* **50**, 3987–4014 (1994).

35. C. Holzhey, F. Larsen, F. Wilczek, "Geometric and renormalized entropy in conformal field theory", *Nuclear Physics B* **424**, 443–467 (1994).

36. S. Hawking, J. Maldacena, A. Strominger, "DeSitter entropy, quantum entanglement and ADS/CFT", *J. High Energy Phys.* **2001**, 001 (2001).

37. P. Calabrese, J. Cardy, "Entanglement entropy and quantum field theory", *J. Stat. Mech.* **2004**, P06002 (2004).

38. K. Audenaert, J. Eisert, M. B. Plenio, R. F. Werner, "Entanglement properties of the harmonic chain", *Phys. Rev. A* **66**, 042327 (2002).

39. T. J. Osborne, M. A. Nielsen, "Entanglement in a simple quantum phase transition", *Phys. Rev. A* **66**, 032110 (2002).

40. A. Osterloh, L. Amico, G. Falci, R. Fazio, "Scaling of entanglement close to a quantum phase transition", *Nature* **416**, 608–610 (2002).

41. M. B. Hastings, "An area law for one-dimensional quantum systems", **2007**, P08024 (2007).

42. E. Witten, "Anti De Sitter Space And Holography", *Adv. Theor. Math. Phys.* **2**, 253 (1998).

43. O. Aharony, S. S. Gubser, J. Maldacena, H. Ooguri, Y. Oz, "Large N field theories, string theory and gravity", *Physics Reports* **323**, 183–386 (2000).

44. G. 't Hooft, "On the quantum structure of a black hole", *Nuclear Physics B* **256**, 727–745 (1985).

45. L. Susskind, "The world as a hologram", *Journal of Mathematical Physics* **36**, 6377–6396 (1995).

46. I. Montvay, G. Münster, *Quantum fields on a lattice* (Univ. Press, Cambridge, 1st paperback ed, 1997), 491 pp.

47. L. Amico, R. Fazio, A. Osterloh, V. Vedral, "Entanglement in many-body systems", *Rev. Mod. Phys.* **80**, 517–576 (2008).

48. N. Schuch, M. M. Wolf, F. Verstraete, J. I. Cirac, "Entropy Scaling and Simulability by Matrix Product States", *Phys. Rev. Lett.* **100**, 030504 (2008).

49. D. N. Page, "Average entropy of a subsystem", *Phys. Rev. Lett.* **71**, 1291–1294 (1993).

50. D. M. Greenberger, M. A. Horne, A. Zeilinger, "Going Beyond Bell's Theorem", arXiv: `quant-ph/0712.0921` (2007).

51. I. Affleck, T. Kennedy, E. H. Lieb, H. Tasaki, "Rigorous results on valence-bond ground states in antiferromagnets", *Phys. Rev. Lett.* **59**, 799–802 (1987).

52. M. Fannes, B. Nachtergaele, R. F. Werner, "Finitely correlated states on quantum spin chains", *Comm. Math. Phys.* **144**, 443–490 (1992).

53. F. Verstraete, D. Porras, J. I. Cirac, "Density Matrix Renormalization Group and Periodic Boundary Conditions: A Quantum Information Perspective", *Phys. Rev. Lett.* **93**, 227205 (2004).

54. F. Verstraete, J. I. Cirac, "Renormalization algorithms for Quantum-Many Body Systems in two and higher dimensions", arXiv: `cond-mat/0407066` (2004).

55. B. Pirvu, V. Murg, J. I. Cirac, F. Verstraete, "Matrix product operator representations", *New J. Phys.* **12**, 025012 (2010).

56. I. P. McCulloch, "From density-matrix renormalization group to matrix product states", **2007**, P10014–P10014 (2007).

57. S. R. White, "Density matrix formulation for quantum renormalization groups", *Phys. Rev. Lett.* **69**, 2863–2866 (1992).

58. G. Vidal, "Efficient Classical Simulation of Slightly Entangled Quantum Computations", *Phys. Rev. Lett.* **91**, 147902 (2003).

59. S. R. White, A. E. Feiguin, "Real-Time Evolution Using the Density Matrix Renormalization Group", *Phys. Rev. Lett.* **93**, 076401 (2004).

60. E. M. Stoudenmire, S. R. White, "Minimally entangled typical thermal state algorithms", *New J. Phys.* **12**, 055026 (2010).

61. J. C. Mason, D. C. Handscomb, *Chebyshev polynomials* (Chapman & Hall/CRC, Boca Raton, Fla, 2003), 341 pp.

62. P. Pfeuty, "The one-dimensional Ising model with a transverse field", *Annals of Physics* **57**, 79–90 (1970).

63. L. Hogben, Ed., *Handbook of linear algebra* (Chapman & Hall/CRC, Boca Raton, 2007).

No real vectors were harmed in the making of this thesis.